Methods for Convex and General Quadratic Programming*

Philip E. Gill† Elizabeth Wong†

April 29, 2013

Abstract

Computational methods are considered for finding a point that satisfies the second-order necessary conditions for a general (possibly nonconvex) quadratic program (QP). The first part of the paper considers the formulation and analysis of an active-set method for a generic QP with both equality and inequality constraints. The method uses a search direction that is the solution of an equality-constrained subproblem involving a "working set" of linearly independent constraints. The method is a reformulation of a method for general QP first proposed by Fletcher, and modified subsequently by Gould. The reformulation facilitates a simpler analysis and has the benefit that the algorithm reduces to a variant of the simplex method when the QP is a linear program. The search direction is computed from a KKT system formed from the QP Hessian and the gradients of the working-set constraints. It is shown that, under certain circumstances, the solution of this KKT system may be updated using a simple recurrence relation, thereby giving a significant reduction in the number of KKT systems that need to be solved.

The second part of the paper focuses on the solution of QP problems with constraints in so-called standard form. We describe how the constituent KKT systems are solved, and discuss how an initial basis is defined. Numerical results are presented for all quadratic programs in the CUTEr test collection.

Key words. Large-scale quadratic programming, active-set methods, convex and nonconvex quadratic programming, KKT systems, Schur-complement method, variable-reduction method.

*Research supported in part by National Science Foundation grants DMS-0511766 and DMS-0915220, and by Department of Energy grant DE-SC0002349.

†Department of Mathematics, University of California, San Diego, La Jolla, CA 92093-0112 (pgill@ucsd.edu, elwong@ucsd.edu).

1. Introduction

A *quadratic program* (QP) involves the minimization or maximization of a quadratic objective function subject to linear equality and inequality constraints on the variables. Quadratic programs arise in many areas, including economics, applied science and engineering. Important applications include portfolio analysis, support vector machines, structural analysis and optimal control. Quadratic programming also forms a principal computational component of many sequential quadratic programming methods for nonlinear programming (for a recent survey, see Gill and Wong [35]). Interior methods and active-set methods are two alternative approaches to handling the inequality constraints of a quadratic program. In this paper we focus on active-set methods, which have the property that they are able to capitalize on a good estimate of the solution. In particular, if a sequence of related QPs must be solved, then the solution of one problem may be used to "warm start" the next, which can significantly reduce the amount of computation time. This feature makes active-set quadratic programming methods particularly effective in the final stages of sequential quadratic programming method.

In the first part of the paper (comprising Sections 2 and 3), we consider the formulation and analysis of an active-set method for a generic QP of the form

$$\begin{aligned} \underset{x\in\mathbb{R}^n}{\text{minimize}} \quad & \varphi(x) = c^T x + \tfrac{1}{2} x^T H x \\ \text{subject to} \quad & Ax = b, \quad Dx \geq f, \end{aligned} \tag{1.1}$$

where A, b, c, D, f and H are constant, H is symmetric, A is $m \times n$, and D is $m_D \times n$. (In order to simplify the notation, it is assumed that the inequalities involve only lower bounds. However, the method to be described can be generalized to treat all forms of linear constraints.) No assumptions are made about H (other than symmetry), which implies that the objective function $\varphi(x)$ need not be convex. In the nonconvex case, however, convergence will be to local minimizers only. The method under consideration defines a primal-dual search pair associated with the solution of an equality-constrained subproblem involving a "working set" of linearly independent constraints. Unlike existing quadratic programming methods, the working set may include constraints that need not be active at the current iterate. In this context, we reformulate a method for a general QP that was first proposed by Fletcher [20], and modified subsequently by Gould [38]. In this reformulation, the primal-dual search directions satisfy a KKT system of equations formed from the Hessian H and the gradients of the constraints in the working set. The working set is specified by an active-set strategy that controls the inertia (i.e., the number of positive, negative and zero eigenvalues) of the KKT matrix. It is shown in Section 3 that this inertia-controlling strategy guarantees that each set of KKT equations is well-defined and nonsingular. In addition, it is shown that, under certain circumstances, the solution of this KKT system may be updated using a simple recurrence relation, thereby giving a significant reduction in the number of KKT systems that need to be solved. (For conventional inertia-controlling methods that use a working set of active constraints, see, e.g., Gill and Murray [25], and Gill, Murray, Saunders and Wright [32, 33].)

Sections 4–7 form the second part of the paper, which focuses on a method for QPs with constraints written in standard form, which is an example of the generic form (1.1) where the inequalities are the nonnegativity constraints $x \geq 0$. It is shown that if $H = 0$ (so that the problem has a linear objective), then the method is equivalent to a variant of the primal simplex method in which the π-values and reduced costs are updated at each iteration. Section 5 describes two approaches for solving the KKT systems. The first approach is the well-known *variable-reduction method*, which is suitable for problems for which the number of active constraints is comparable to the number of variables (i.e., for problems with a small number of degrees of freedom). The variable reduction method uses a Cholesky factorization of the reduced Hessian and a sparse LU factorization of a basis matrix. The second approach, which we call the *block-LU method*, uses a sparse factorization of a fixed indefinite KKT matrix in conjunction with the factorization of a smaller dense matrix that is updated at each iteration. The use of a fixed factorization allows a "black-box" sparse equation solver to be used repeatedly. This feature makes the block-LU method ideally suited to problems with structure that can be exploited by using specialized factorization. Moreover, improvements in efficiency derived from exploiting new parallel and vector computer architectures are immediately applicable via state-of-the-art linear equation solvers. Section 6 describes how an appropriate initial basis is found when the problem is not strictly convex. Finally, in Section 7 we describe the main features of the Fortran 2003 package `SQIC` (Sparse Quadratic programming using Inertia Control), which is a particular implementation of the method for standard form QPs described in Section 4. Numerical results are given for all the linear and quadratic programs in the CUTEr test collection (see [5, 39]).

Not all active-set methods for a general QP are inertia controlling—see, for example, the methods of Bunch and Kaufman [8], Friedlander and Leyffer [22], and the quadratic programming methods in the `GALAHAD` software package of Gould, Orban, and Toint [41, 42, 40]. A number of alternative methods have been proposed for strictly convex quadratic programming with a modest number of constraints and variables, see, e.g., Goldfarb and Idnani [36], Gill et al. [24], and Powell [50]. A variable-reduction method for a large-scale convex QP is proposed by Gill, Murray and Saunders [26]. Bartlett and Biegler [3] propose a fixed-factorization method for large-scale strictly convex problems (see Section 5.2).

Notation. The gradient of the objective φ evaluated at x, $c + Hx$, is denoted by the vector $g(x)$, or g if it is clear where the evaluation occurs. The vector d_i^T refers to the i-th row of the constraint matrix D, so that the i-th inequality constraint is $d_i^T x \geq f_i$. The i-th component of a vector labeled with a subscript will be denoted by $[\,\cdot\,]_i$, e.g., $[v_N]_i$ is the i-th component of the vector v_N. Similarly, a subvector of components with indices in the index set $\mathcal{S}$ is denoted by $(\,\cdot\,)_{\mathcal{S}}$, e.g., $(v_N)_{\mathcal{S}}$ is the vector with components $[v_N]_i$ for $i \in \mathcal{S}$. The symbol I is used to denote an identity matrix with dimension determined by the context. The j-th column of I is denoted by e_j. Unless explicitly indicated otherwise, $\|\cdot\|$ denotes the vector two-norm or its induced matrix norm. The inertia of a real symmetric matrix A, denoted by $\mathrm{In}(A)$, is the integer triple (a_+, a_-, a_0) giving the number of positive, negative and zero eigenvalues of A. Given vectors a and b with the same dimension, the vector with i-th component $a_i b_i$ is denoted by $a \cdot b$. Given symmetric $K = \begin{pmatrix} M & N^T \\ N & G \end{pmatrix}$, with M nonsingular, the matrix $G - NM^{-1}N^T$, the Schur complement of M in K, will be denoted by K/M. When the definitions of the relevant matrices are clear we will refer to "the" Schur complement.

2. Background

In this section, we review the optimality conditions for the generic QP (1.1), and describe a framework for the formulation of feasible-point active-set QP methods. Throughout, it is assumed that the matrix A has full row-rank m. This condition is easily satisfied for the class of active-set methods considered in this paper. Given an arbitrary matrix G, equality constraints $Gu = b$ are equivalent to the full rank constraints $Gu + v = b$, if we impose $v = 0$. In this formulation, the v-variables are artificial variables that are fixed at zero.

2.1. Optimality conditions

The necessary and sufficient conditions for a local solution of the QP (1.1) involve the existence of vectors z and π of Lagrange multipliers associated with the constraints $Dx \geq f$ and $Ax = b$, respectively. The conditions are summarized by the following result, which is stated without proof (see, e.g., Borwein [6], Contesse [9] and Majthay [46]).

Result 2.1. (QP optimality conditions) *The point x is a local minimizer of the quadratic program (1.1) if and only if*

(a) *$Ax = b$, $Dx \geq f$, and there exists at least one pair of vectors π and z such that $g(x) = A^T\pi + D^Tz$, with $z \geq 0$, and $z \cdot (Dx - f) = 0$;*

(b) *$p^THp \geq 0$ for all nonzero p satisfying $g(x)^Tp = 0$, $Ap = 0$, and $d_i^Tp \geq 0$ for every i such that $d_i^Tx = f_i$.* ∎

We follow the convention of refering to any x that satisfies condition (a) as a first-order KKT point.

If H has at least one negative eigenvalue and (x, π, z) satisfies condition (a) with an index i such that $z_i = 0$ and $d_i^Tx = f_i$, then x is known as a dead point. Verifying condition (b) at a dead point requires finding the global minimizer of an indefinite quadratic form over a cone, which is an NP-hard problem (see, e.g., Cottle, Habetler and Lemke [10], Pardalos and Schnitger [48], and Pardalos and Vavasis [49]). This implies that the optimality of a candidate solution of a general quadratic program can be verified only if more restrictive (but computationally tractable) sufficient conditions are satisfied. A dead point is a point at which the sufficient conditions are not satisfied, but certain necessary conditions for optimality hold. Replacing part (b) of Result 2.1 with the condition that $p^THp \geq 0$ for all nonzero p satisfying $Ap = 0$, and $d_i^Tp = 0$ for each i such that $d_i^Tx = f_i$, leads to computationally tractable necessary conditions for optimality.

Additionally, suitable sufficient conditions for optimality are given by replacing the necessary condition by the condition that $p^THp \geq 0$ for all p such that $Ap = 0$, and $d_i^Tp = 0$ for every $i \in \mathcal{A}_+(x)$, where $\mathcal{A}_+(x)$ is the index set $\mathcal{A}_+(x) = \{i : d_i^Tx = f_i \text{ and } z_i > 0\}$.

These conditions may be expressed in terms of the constraints that are satisfied with equality at x. Let x be any point satisfying the equality constraints $Ax = b$. (The assumption that A has rank m implies that there must exist at least one such x.) An inequality constraint is active at x if it is satisfied with equality. The indices associated with the active constraints comprise the active set, denoted by $\mathcal{A}(x)$. An active-constraint matrix $A_{\mathfrak{a}}(x)$ is a matrix with rows consisting of the rows of A and the gradients of the active constraints. By convention, the rows of A are listed first, giving the active-constraint matrix

$$A_{\mathfrak{a}}(x) = \begin{pmatrix} A \\ D_{\mathfrak{a}}(x) \end{pmatrix},$$

where $D_{\mathfrak{a}}(x)$ comprises the rows of D with indices in $\mathcal{A}(x)$. Note that the active-constraint matrix includes A in addition to the gradients of the active constraints. The argument x is generally omitted if it is clear where $D_{\mathfrak{a}}$ is defined.

With this definition of the active set, we give necessary conditions for the QP.

Result 2.2. (Necessary conditions in active-set form) *Let the columns of the matrix $Z_{\mathfrak{a}}$ form a basis for the null-space of $A_{\mathfrak{a}}$. The point x is a local minimizer of the QP (1.1) only if*

(a) *x is a first-order KKT point, i.e.,* (i) $Ax = b$, $Dx \geq f$; (ii) $g(x)$ *lies in* $\operatorname{range}(A_{\mathfrak{a}}^T)$, *or equivalently, there exist vectors* π *and* $z_{\mathfrak{a}}$ *such that* $g(x) = A^T\pi + D_{\mathfrak{a}}^T z_{\mathfrak{a}}$; *and* (iii) $z_{\mathfrak{a}} \geq 0$,

(b) *the reduced Hessian* $Z_{\mathfrak{a}}^T H Z_{\mathfrak{a}}$ *is positive semidefinite.* ∎

Typically, software for general quadratic programming will terminate the iterations at a dead point. Nevertheless, it is possible to define procedures that check for optimality at a dead point, even though the chance of success in a reasonable amount of computation time will depend on the size of the problem (see Forsgren, Gill and Murray [21]).

2.2. Active-set methods

The method to be considered is a two-phase active-set method. In the first phase (the feasibility phase or phase 1), the objective is ignored while a feasible point is found for the constraints $Ax = b$ and $Dx \geq f$. In the second phase (the optimality phase or phase 2), the objective is minimized while feasibility is maintained. Given a feasible x_0, active-set methods compute a sequence of feasible iterates $\{x_k\}$ such that $x_{k+1} = x_k + \alpha_k p_k$ and $\varphi(x_{k+1}) \leq \varphi(x_k)$, where p_k is a nonzero search direction and α_k is a nonnegative step length. Active-set methods are motivated by the main result of Farkas' Lemma, which states that a feasible x must either satisfy the first-order optimality conditions or be the starting point of a feasible descent direction, i.e., a direction p such that

$$A_{\mathfrak{a}} p \geq 0 \quad \text{and} \quad g(x)^T p < 0. \tag{2.1}$$

The method considered in this paper approximates the active set by a working set $\mathcal{W}$ of row indices of D. The working set has the form $\mathcal{W} = \{\nu_1, \nu_2, \ldots, \nu_{m_w}\}$, where m_w is the number of indices in $\mathcal{W}$. Analogous to the active-constraint matrix $A_{\mathfrak{a}}$, the $(m+m_w)\times n$ working-set matrix A_w contains the gradients of the equality constraints and inequality constraints in $\mathcal{W}$. The structure of the working-set matrix is similar to that of the active-set matrix, i.e.,

$$A_w = \begin{pmatrix} A \\ D_w \end{pmatrix},$$

where D_w is a matrix formed from the m_w rows of D with indices in $\mathcal{W}$. The vector f_w denotes the components of f with indices in $\mathcal{W}$.

There are two important distinctions between the definitions of $\mathcal{A}$ and $\mathcal{W}$.

(i) The indices of $\mathcal{W}$ define a subset of the rows of D that are linearly independent of the rows of A, i.e., the working set matrix A_w has full row rank. It follows that m_w must satisfy $0 \leq m_w \leq \min\{n-m, m_D\}$.

(ii) The active set $\mathcal{A}$ is uniquely defined at any feasible x, whereas there may be many choices for $\mathcal{W}$. The set $\mathcal{W}$ is determined by the properties of a particular active-set method.

Conventional active-set methods define the working set as a subset of the active set (see, e.g., Gill, Murray and Wright [34], and Nocedal and Wright [47]). In this paper we relax this requirement—in particular, a working-set constraint may be strictly satisfied at x. (More generally, a working-set constraint may be violated at x, although this property is not used here).

Given a working set $\mathcal{W}$ and an associated working-set matrix A_w at x, we introduce the notions of stationarity and optimality with respect to a working set. We emphasize that the definitions below do not require that the working-set constraints are active (or even feasible) at x.

Definition 2.1. (Subspace stationary point) *Let $\mathcal{W}$ be a working set defined at an x such that $Ax = b$. Then x is a* subspace stationary point with respect to $\mathcal{W}$ (or, equivalently, with respect to A_w) *if $g \in \operatorname{range}(A_w^T)$, i.e., there exists a vector y such that $g = A_w^T y$. Equivalently, x is a subspace stationary point with respect to the working set $\mathcal{W}$ if the reduced gradient $Z_w^T g$ is zero, where the columns of Z_w form a basis for the null-space of A_w.* ∎

At a subspace stationary point, the components of y are the Lagrange multipliers associated with a QP with equality constraints $Ax = b$ and $D_w x = f_w$. To be consistent with the optimality conditions of Result 2.2, we denote the first m components of y as π (the multipliers associated with $Ax = b$) and the last m_w components of y as z_w (the multipliers associated with the constraints in $\mathcal{W}$). With this notation, the identity $g(x) = A_w^T y = A^T\pi + D_w^T z_w$ holds at a subspace stationary point.

To classify subspace stationary points based on curvature information, we define the terms *second-order-consistent working set* and *subspace minimizer*.

Definition 2.2. (Second-order-consistent working set) *Let $\mathcal{W}$ be a working set associated with an x such that $Ax = b$, and let the columns of Z_w form a basis for the null-space of A_w. The working set $\mathcal{W}$ is* second-order-consistent *if the reduced Hessian $Z_w^T H Z_w$ is positive definite.* ∎

The inertia of the reduced Hessian is related to the inertia of the $(n+m+m_w)\times(n+m+m_w)$ KKT matrix $K = \begin{pmatrix} H & A_w^T \\ A_w & \end{pmatrix}$ through the identity $\operatorname{In}(K) = \operatorname{In}(Z_w^T H Z_w) + (m+m_w, m+m_w, 0)$ (see Gould [37]). It follows that an equivalent characterization of a second-order-consistent working set is that K has inertia $(n, m+m_w, 0)$. A KKT matrix K associated with a second-order-consistent working set is said to have "*correct inertia*". It is always possible to impose sufficiently many *temporary constraints* that will covert a given working set into a second-order consistent working set. For example, a temporary vertex formed by fixing variables at their current values will always provide a KKT matrix with correct inertia (see Section 6 for more details).

Definition 2.3. (Subspace minimizer) *If x is a subspace stationary point with respect to a second-order-consistent basis $\mathcal{W}$, then x is known as a* subspace minimizer with respect to $\mathcal{W}$. *If every constraint in the working set is active, then x is called a* standard subspace minimizer*; otherwise x is called a* nonstandard subspace minimizer*.* ∎

3. A Method for the Generic Quadratic Program

In this section we formulate and analyze an active-set method based on controlling the inertia of the KKT matrix. Inertia-controlling methods were first proposed by Fletcher [20] and are based on the simple rule that a constraint is removed from the working set only at a *subspace minimizer*. We show that with an appropriate choice of initial point, this rule ensures that every iterate is a subspace minimizer for the associated working set. This allows for the reliable and efficient calculation of the search directions.

The method starts at a subspace minimizer x with $g(x) = A_w^T y = A^T\pi + D_w^T z_w$ and a KKT matrix with correct inertia. If x is standard and $z_w \geq 0$, then x is optimal for the QP. Otherwise, there exists an index $\nu_s \in \mathcal{W}$ such that $[z_w]_s < 0$. To proceed, we define a descent direction that is feasible for the equality constraints and the constraints in the working set. Analogous to (2.1), p is defined so that

$$A_w p = e_{m+s} \quad \text{and} \quad g(x)^T p < 0.$$

We call any vector satisfying this condition a *nonbinding direction* because any nonzero step along it will increase the residual of the ν_s-th inequality constraint (and hence make it inactive or nonbinding). Here we define p as the solution of the equality-constrained subproblem

$$\underset{p}{\text{minimize}} \;\; \varphi(x+p) \quad \text{subject to} \quad A_w p = e_{m+s}. \tag{3.1}$$

The optimality conditions for this subproblem imply the existence of a vector q such that $g(x+p) = A_w^T(y+q)$; i.e., q is the step to the multipliers associated with the optimal solution $x+p$. This condition, along with the feasibility condition, implies that p and q satisfy the equations

$$\begin{pmatrix} H & A_w^T \\ A_w & 0 \end{pmatrix} \begin{pmatrix} p \\ -q \end{pmatrix} = \begin{pmatrix} -(g(x) - A_w^T y) \\ e_{m+s} \end{pmatrix}. \tag{3.2}$$

The primal and dual vectors have a number of important properties that are summarized in the next result.

Result 3.1. (Properties of the search direction) *Let x be a subspace minimizer such that $g = A_w^T y = A^T\pi + D_w^T z_w$, with $[z_w]_s < 0$. Then the vectors p and q satisfying the equations*

$$\begin{pmatrix} H & A_w^T \\ A_w & 0 \end{pmatrix} \begin{pmatrix} p \\ -q \end{pmatrix} = \begin{pmatrix} -(g(x) - A_w^T y) \\ e_{m+s} \end{pmatrix} = \begin{pmatrix} 0 \\ e_{m+s} \end{pmatrix} \tag{3.3}$$

constitute the unique primal and dual solutions of the equality constrained problem defined by minimizing $\varphi(x+p)$ subject to $A_w p = e_{m+s}$. Moreover, p and q satisfy the identities

$$g^T p = y_{m+s} = [z_w]_s \quad \text{and} \quad p^T H p = q_{m+s} = [q_w]_s, \tag{3.4}$$

where q_w denotes the vector of last m_w components of q.

Proof. The assumption that x is a subspace minimizer implies that the subproblem has a unique bounded minimizer. The optimality of p and q follows from the equations in (3.2), which represent the feasibility and optimality conditions for the minimization of $\varphi(x+p)$ on the set $\{p : A_w p = e_{m+s}\}$. The equation $g = A_w^T y$ and the definition of p from (3.3) give

$$g^T p = p^T(A_w^T y) = y^T A_w p = y^T e_{m+s} = y_{m+s} = [z_w]_s$$

Similarly, $p^T H p = p^T(A_w^T q) = e_{m+s}^T q = q_{m+s} = [q_w]_s$. ∎

Once p and q are known, a nonnegative step α is computed so that $x + \alpha p$ is feasible and $\varphi(x+\alpha p) \le \varphi(x)$. If $p^T H p > 0$, the step that minimizes $\varphi(x+\alpha p)$ as a function of α is given by $\alpha_* = -g^T p / p^T H p$. The identities (3.4) give

$$\alpha_* = -g^T p / p^T H p = -[z_w]_s / [q_w]_s.$$

Since $[z_w]_s < 0$, if $[q_w]_s = p^T H p > 0$, the optimal step α_* is positive. Otherwise $[q_w]_s = p^T H p \le 0$ and φ has no bounded minimizer along p and $\alpha_* = +\infty$.

If $x + \alpha_* p$ is unbounded or infeasible, then α must be limited by α_F, the *maximum feasible step* from x along p. The feasible step is defined as $\alpha_F = \gamma_r$, where

$$\gamma_r = \min \gamma_i, \quad \text{with} \quad \gamma_i = \begin{cases} \dfrac{d_i^T x - f_i}{-\,d_i^T p} & \text{if } d_i^T p < 0; \\ +\infty & \text{otherwise.} \end{cases}$$

The step α is then $\min\{\alpha_*, \alpha_F\}$. If $\alpha = +\infty$, the QP has no bounded solution and the algorithm terminates. In the discussion below, we assume that α is a bounded step.

The primal and dual directions p and q defined by (3.3) have the property that $x + \alpha p$ remains a subspace minimizer with respect to A_w *for any step* α. This follows from the definitions (3.3), which imply that

$$g(x + \alpha p) = g(x) + \alpha Hp = A_w^T y + \alpha A_w^T q = A_w^T(y + \alpha q), \tag{3.5}$$

so that the gradient at $x+\alpha p$ is a linear combination of the columns of A_w^T. The step $x+\alpha p$ does not change the KKT matrix K associated with the subspace minimizer x, which implies that $x + \alpha p$ is also a subspace minimizer with respect to A_w. This means that $x + \alpha p$ may be interpreted as the solution of a problem in which the working-set constraint $d_{\nu_s}^T x \geq f_{\nu_s}$ is *shifted* to pass through $x + \alpha p$. The component $[y + \alpha q]_{m+s} = [z_w + \alpha q_w]_s$ is the Lagrange multiplier associated with the shifted version of $d_{\nu_s}^T x \geq f_{\nu_s}$. This property is known as the *parallel subspace property* of quadratic programming. It shows that if x is stationary with respect to a nonbinding constraint, then it remains so for all subsequent iterates for which that constraint remains in the working set.

Once α has been defined, the new iterate is $\bar{x} = x + \alpha p$. The composition of the new working set and multipliers depends on the definition of α.

Case 1: $\alpha = \alpha_*$ In this case, the step $\alpha = \alpha_* = -[z_w]_s/[q_w]_s$ minimizes $\varphi(x + \alpha p)$ with respect to α, giving the s-th element of $z_w + \alpha q_w$ as

$$[z_w + \alpha q_w]_s = [z_w]_s + \alpha_*[q_w]_s = 0,$$

which implies that the Lagrange multiplier associated with the shifted constraint is *zero* at $\bar{x}$. The nature of the stationarity may be determined using the next result.

Result 3.2. (Constraint deletion) *Let x be a subspace minimizer with respect to $\mathcal{W}$. Assume that $[z_w]_s < 0$. Let $\bar{x}$ denote the point $x + \alpha p$, where p is defined by (3.3) and $\alpha = \alpha_*$ is bounded. Then $\bar{x}$ is a subspace minimizer with respect to $\bar{\mathcal{W}} = \mathcal{W} - \{\nu_s\}$.*

Proof. Let K and $\bar{K}$ denote the matrices

$$K = \begin{pmatrix} H & A_w^T \\ A_w & \end{pmatrix} \quad \text{and} \quad \bar{K} = \begin{pmatrix} H & \bar{A}_w^T \\ \bar{A}_w & \end{pmatrix},$$

where A_w and $\bar{A}_w$ are the working-set matrices associated with $\mathcal{W}$ and $\bar{\mathcal{W}}$. It suffices to show that $\bar{K}$ has the correct inertia, i.e., $\mathrm{In}(\bar{K}) = (n, m + m_w - 1, 0)$.

Consider the matrix M such that

$$M \triangleq \begin{pmatrix} K & e_{m+n+s} \\ e_{m+n+s}^T & \end{pmatrix}.$$

By assumption, x is a subspace minimizer with $\mathrm{In}(K) = (n, m + m_w, 0)$. In particular, K is nonsingular and the Schur complement of K in M exists with

$$M/K = -e_{n+m+s}^T K^{-1} e_{n+m+s} = -e_{n+m+s}^T \begin{pmatrix} p \\ -q \end{pmatrix} = [q_w]_s.$$

It follows that

$$\mathrm{In}(M) = \mathrm{In}(M/K) + \mathrm{In}(K) = \mathrm{In}([q_w]_s) + (n, m + m_w, 0). \tag{3.6}$$

Now consider a symmetrically permuted version of M:

$$\widetilde{M} = \begin{pmatrix} 0 & 1 & & \\ 1 & 0 & d_{\nu_s}^T & \\ & d_{\nu_s} & H & \bar{A}_w^T \\ & & \bar{A}_w & \end{pmatrix}.$$

Inertia is unchanged by symmetric permutations, so $\text{In}(M) = \text{In}(\widetilde{M})$. The 2×2 block in the upper-left corner of $\widetilde{M}$, denoted by E, has eigenvalues ± 1, so that

$$\text{In}(E) = (1, 1, 0) \quad \text{with} \quad E^{-1} = E.$$

The Schur complement of E in $\widetilde{M}$ is

$$\widetilde{M}/E = \bar{K} - \begin{pmatrix} 0 & d_{\nu_s} \\ 0 & 0 \end{pmatrix} \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix} \begin{pmatrix} 0 & 0 \\ d_{\nu_s}^T & 0 \end{pmatrix} = \bar{K},$$

which implies that $\text{In}(\widetilde{M}) = \text{In}(\widetilde{M}/E) + \text{In}(E) = \text{In}(\bar{K}) + (1, 1, 0)$. Combining this with (3.6) yields

$$\begin{aligned} \text{In}(\bar{K}) &= \text{In}([q_w]_s) + (n, m + m_w, 0) - (1, 1, 0) \\ &= \text{In}([q_w]_s) + (n - 1, m + m_w - 1, 0). \end{aligned}$$

As $\alpha = \alpha_*$, the scalar $[q_w]_s$ must be positive. It follows that

$$\text{In}(\bar{K}) = (1, 0, 0) + (n - 1, m + m_w - 1, 0) = (n, m + m_w - 1, 0)$$

and the subspace stationary point $\bar{x}$ is a (standard) subspace minimizer with respect to the new working set $\bar{\mathcal{W}} = \mathcal{W} - \{\nu_s\}$. ∎

Case 2: $\alpha = \alpha_F$ In this case, α is the step to the blocking constraint $d_r^T x \geq f_r$, which is eligible to be added to the working set at $x + \alpha p$. However, the definition of the new working set depends on whether or not the blocking constraint is dependent on the constraints already in $\mathcal{W}$. If d_r is linearly independent of the columns of A_w^T, then the index r is added to the working set. Otherwise, we show in Result 3.4 below that a suitable working set is defined by exchanging rows d_{ν_s} and d_r in A_w. The following result provides a computable test for the independence of d_r and the columns of A_w^T.

Result 3.3. (Test for constraint dependency) *Let x be a subspace minimizer with respect to A_w. Assume that $d_r^T x \geq f_r$ is a blocking constraint at $\bar{x} = x + \alpha p$, where p satisfies (3.3). Define vectors u and v such that*

$$\begin{pmatrix} H & A_w^T \\ A_w & \end{pmatrix} \begin{pmatrix} u \\ v \end{pmatrix} = \begin{pmatrix} d_r \\ 0 \end{pmatrix}, \tag{3.7}$$

then

(a) *d_r and the columns of A_w^T are linearly independent if and only if $u \neq 0$;*

(b) *$v_{m+s} = d_r^T p < 0$, and $u^T d_r \geq 0$ with $u^T d_r > 0$ if $u \neq 0$.*

Proof. For part (a), equations (3.7) give $Hu + A_w^T v = d_r$ and $A_w u = 0$. If $u = 0$ then $A_w^T v = d_r$, and d_r must be dependent on the columns of A_w^T. Conversely, if $A_w^T v = d_r$, then the definition of u gives $u^T A_w^T v = u^T d_r = 0$, which implies that $u^T H u = u^T(Hu + A_w^T v) = u^T d_r = 0$. By assumption, x is a subspace minimizer with respect to A_w, which is equivalent to the assumption that H is positive definite for all u such that $A_w u = 0$. Hence $u^T H u = 0$ can hold only if u is zero.

For part (b), we use equations (3.3) and (3.7) to show that

$$v_{m+s} = e_{m+s}^T v = p^T A_w^T v = p^T(d_r - Hu) = p^T d_r - q^T A_w u = d_r^T p < 0,$$

where the final inequality follows from the fact that $d_r^T p$ must be negative if $d_r^T x \geq f_r$ is a blocking constraint. If $u \neq 0$, equations (3.7) imply $Hu + A_w^T v = d_r$ and $A_w u = 0$. Multiplying the first equation by u^T and applying the second equation gives $u^T H u = u^T d_r$. As $u \in \text{null}(A_w)$ and x is a subspace minimizer, it must hold that $u^T H u = u^T d_r > 0$, as required. ∎

The next result provides expressions for the updated multipliers.

Result 3.4. (Multiplier updates) *Assume that x is a subspace minimizer with respect to A_w. Assume that $d_r^T x \geq f_r$ is a blocking constraint at the next iterate $\bar{x} = x + \alpha p$, where the direction p satisfies (3.3). Let u and v satisfy (3.7).*

(a) *If d_r and the columns of A_w^T are linearly independent, then the vector $\bar{y}$ formed by appending a zero component to the vector $y + \alpha q$ satisfies $g(\bar{x}) = \bar{A}_w^T \bar{y}$, where $\bar{A}_w$ denotes the matrix A_w with row d_r^T added in the last position.*

(b) *If d_r and the columns of A_w^T are linearly dependent, then the vector $\bar{y}$ such that*

$$\bar{y} = y + \alpha q - \sigma v, \quad \text{with} \quad \sigma = [y + \alpha q]_{m+s} / v_{m+s}, \tag{3.8}$$

satisfies $g(\bar{x}) = A_w^T \bar{y} + \sigma d_r$ with $\bar{y}_{m+s} = 0$ and $\sigma > 0$.

Proof. For part (a), the parallel subspace property (3.5) implies that $g(x + \alpha p) = g(\bar{x}) = A_w^T(y + \alpha q)$. As d_r and the columns of A_w^T are linearly independent, we may add the index r to $\mathcal{W}$ and define the new working-set matrix $\bar{A}_w^T = \begin{pmatrix} A_w^T & d_r \end{pmatrix}$. This allows us to write $g(\bar{x}) = \bar{A}_w^T \bar{y}$, with $\bar{y}$ given by $y + \alpha q$ with an appended zero component.

Now assume that A_w^T and d_r are linearly dependent. From Result 3.3 it must hold that $u = 0$ and there exists a unique v such that $d_r = A_w^T v$. For any value of σ, the parallel subspace property (3.5) gives

$$g(\bar{x}) = A_w^T(y + \alpha q) = A_w^T(y + \alpha q - \sigma v) + \sigma d_r.$$

If we choose $\sigma = [y + \alpha q]_{m+s} / v_{m+s}$ and define the vector $\bar{y} = y + \alpha q - \sigma v$, then

$$g(\bar{x}) = A_w^T \bar{y} + \sigma d_r, \quad \text{with} \quad \bar{y}_{m+s} = [y + \alpha q - \sigma v]_{m+s} = 0.$$

It follows that $g(\bar{x})$ is a linear combination of d_r and every column of A_w^T except d_s.

In order to show that $\sigma = [y + \alpha q]_{m+s} / v_{m+s}$ is positive, we consider the linear function $y_{m+s}(\alpha) = [y + \alpha q]_{m+s}$, which satisfies $y_{m+s}(0) = y_{m+s} < 0$. If $q_{m+s} = p^T H p > 0$, then $\alpha_* < \infty$ and $y_{m+s}(\alpha)$ is an increasing linear function of positive α with $y_{m+s}(\alpha_*) = 0$. This implies that $y_{m+s}(\alpha) < 0$ for any $\alpha < \alpha_*$ and $y_{m+s}(\alpha_k) < 0$. If $q_{m+s} \leq 0$, then $y_{m+s}(\alpha)$ is a nonincreasing linear function of α so that $y_{m+s}(\alpha) < 0$ for any positive α. Thus, $[y + \alpha q]_{m+s} < 0$ for any $\alpha < \alpha_*$, and $\sigma = [y + \alpha q]_{m+s} / v_{m+s} > 0$ from part (b) of Result 3.3. ∎

Result 3.5. *Let x be a subspace minimizer with respect to the working set $\mathcal{W}$. Assume that $d_r^T x \ge f_r$ is a blocking constraint at $\bar{x} = x + \alpha p$, where p is defined by (3.3).*

(a) *If d_r is linearly independent of the columns of A_w^T, then $\bar{x}$ is a subspace minimizer with respect to the working set $\bar{\mathcal{W}} = \mathcal{W} + \{r\}$.*

(b) *If d_r is dependent on the columns of A_w^T, then $\bar{x}$ is a subspace minimizer with respect to the working set $\bar{\mathcal{W}} = \mathcal{W} + \{r\} - \{\nu_s\}$.*

Proof. Parts (a) and (b) of Result 3.4 imply that $\bar{x}$ is a subspace stationary point with respect to $\bar{\mathcal{W}}$. It remains to show that in each case, the new working sets are second-order-consistent.

For part (a), the new KKT matrix for the new working set $\bar{\mathcal{W}} = \mathcal{W} + \{r\}$ must have inertia $(n, m+m_w+1, 0)$. Assume that d_r and the columns of A_w^T are linearly independent, so that the vector u of (3.7) is nonzero. Let K and $\bar{K}$ denote the KKT matrices associated with the working sets $\mathcal{W}$ and $\bar{\mathcal{W}}$, i.e.,

$$K = \begin{pmatrix} H & A_w^T \\ A_w & \end{pmatrix} \quad \text{and} \quad \bar{K} = \begin{pmatrix} H & \bar{A}_w^T \\ \bar{A}_w & \end{pmatrix},$$

where $\bar{A}_w$ is the matrix A_w with the row d_r^T added in the last position.

By assumption, x is a subspace minimizer and $\mathrm{In}(K) = (n, m+m_w, 0)$. It follows that K is nonsingular and the Schur complement of K in $\bar{K}$ exists with

$$\bar{K}/K = -\begin{pmatrix} d_r \\ 0 \end{pmatrix}^T K^{-1} \begin{pmatrix} d_r \\ 0 \end{pmatrix} = -\begin{pmatrix} d_r^T & 0 \end{pmatrix} \begin{pmatrix} u \\ v \end{pmatrix} = -d_r^T u < 0,$$

where the last inequality follows from part (b) of Result 3.3. Then,

$$\begin{aligned} \mathrm{In}(\bar{K}) = \mathrm{In}(\bar{K}/K) + \mathrm{In}(K) &= \mathrm{In}(-u^T d_r) + (n, m+m_w, 0) \\ &= (0,1,0) + (n, m+m_w, 0) = (n, m+m_w+1, 0). \end{aligned}$$

For part (b), assume that d_r and the columns of A_w^T are linearly dependent and that $\bar{\mathcal{W}} = \mathcal{W} + \{r\} - \{\nu_s\}$. By Result 3.4 and equation (3.7), it must hold that $u = 0$ and $A_w^T v = d_r$. Let A_w and $\bar{A}_w$ be the working-set matrices associated with $\mathcal{W}$ and $\bar{\mathcal{W}}$. The change in the working set replaces row s of D_w by d_r^T, so that

$$\begin{aligned} \bar{A}_w = A_w + e_{m+s}(d_r^T - d_s^T) &= A_w + e_{m+s}(v^T A_w - e_{m+s}^T A_w) \\ &= \big(I_w + e_{m+s}(v - e_{m+s})^T\big) A_w \\ &= M A_w, \end{aligned}$$

where $M = I_w + e_{m+s}(v - e_{m+s})^T$. The matrix M has $m + m_w - 1$ unit eigenvalues and one eigenvalue equal to v_{m+s}. From part (b) of Result 3.3, it holds that $v_{m+s} < 0$ and hence M is nonsingular. The new KKT matrix for $\bar{\mathcal{W}}$ can be written as

$$\begin{pmatrix} H & \bar{A}_w^T \\ \bar{A}_w & \end{pmatrix} = \begin{pmatrix} I_n & \\ & M \end{pmatrix} \begin{pmatrix} H & A_w^T \\ A_w & \end{pmatrix} \begin{pmatrix} I_n & \\ & M^T \end{pmatrix}.$$

By Sylvester's Law of Inertia, the old and new KKT matrices have the same inertia, which implies that $\bar{x}$ is a subspace minimizer with respect to $\bar{\mathcal{W}}$. ∎

The first part of this result shows that $\bar{x}$ is a subspace minimizer both before and after an independent constraint is added to the working set. This is crucial because it means that the directions p and q for the next iteration satisfy the KKT equations (3.3) with $\bar{A}_w$ in place of A_w. The second part shows that the working-set constraints can be linearly dependent only at a *standard* subspace minimizer associated with a working set that does not include constraint ν_s. This implies that it is appropriate to remove ν_s from the working set. The constraint $d_{\nu_s}^T x \geq f_{\nu_s}$ plays a significant (and explicit) role in the definition of the search direction and is called the *nonbinding working-set constraint.* The method generates sets of consecutive iterates that begin and end with a standard subspace minimizer. The nonbinding working-set constraint $d_{\nu_s}^T x \geq f_{\nu_s}$ identified at the first point of the sequence is deleted from the working set at the last point (either by deletion or replacement).

Each iteration requires the solution of two KKT systems:

$$\text{Full System 1} \qquad \begin{pmatrix} H & A_w^T \\ A_w & 0 \end{pmatrix} \begin{pmatrix} p \\ -q \end{pmatrix} = \begin{pmatrix} 0 \\ e_{m+s} \end{pmatrix} \tag{3.9a}$$

$$\text{Full System 2} \qquad \begin{pmatrix} H & A_w^T \\ A_w & 0 \end{pmatrix} \begin{pmatrix} u \\ v \end{pmatrix} = \begin{pmatrix} d_r \\ 0 \end{pmatrix}. \tag{3.9b}$$

However, for those iterations for which the number of constraints in the working set increases, it is possible to *update* the vectors p and q, making it unnecessary to solve (3.9a).

Result 3.6. *Let x be a subspace minimizer with respect to A_w. Assume the vectors p, q, u and v are defined by (3.9). Let d_r be the gradient of a blocking constraint at $\bar{x} = x + \alpha p$ such that d_r is independent of the columns of A_w^T. If $\rho = -d_r^T p / d_r^T u$, then the vectors*

$$\bar{p} = p + \rho u \quad \text{and} \quad \bar{q} = \begin{pmatrix} q - \rho v \\ \rho \end{pmatrix}$$

are well-defined and satisfy

$$\begin{pmatrix} H & \bar{A}_w^T \\ \bar{A}_w & \end{pmatrix} \begin{pmatrix} \bar{p} \\ -\bar{q} \end{pmatrix} = \begin{pmatrix} 0 \\ e_{m+s} \end{pmatrix}, \quad \text{where} \quad \bar{A}_w = \begin{pmatrix} A_w \\ d_r^T \end{pmatrix}. \tag{3.10}$$

Proof. Result 3.3 implies that u is nonzero and that $u^T d_r > 0$ so that ρ is well defined (and strictly positive).

For any scalar ρ, (3.9a) and (3.9b) imply that

$$\begin{pmatrix} H & A_w^T & d_r \\ A_w & & \\ d_r^T & & \end{pmatrix} \begin{pmatrix} p + \rho u \\ -(q - \rho v) \\ -\rho \end{pmatrix} = \begin{pmatrix} 0 \\ e_{m+s} \\ d_r^T p + \rho d_r^T u \end{pmatrix}.$$

If ρ is chosen so that $d_r^T p + \rho d_r^T u = 0$, the last component of the right-hand side is zero, and $\bar{p}$ and $\bar{q}$ satisfy (3.10) as required. ∎

With a suitable nondegeneracy assumption, the algorithm terminates in a finite number of iterations. Since the number of constraints is finite, the sequence $\{x_k\}$ must contain a subsequence $\{x_{ik}\}$ of standard subspace minimizers with respect to their working sets $\{\mathcal{W}_{ik}\}$. If the Lagrange multipliers are nonnegative at any of these points, the algorithm terminates with the desired solution. Otherwise, at least one multiplier must be strictly negative, and hence the nondegeneracy assumption implies that $\alpha_F > 0$ at x_{ik}. Thus, $\varphi(x_{ik}) > \varphi(x_{ik} + \alpha_{ik} p_{ik})$, since at each iteration, the direction is defined as a descent direction with $g^T p < 0$. The subsequence $\{x_{ik}\}$ must be finite because the number of subspace minimizers is finite and the strict decrease in $\varphi(x)$ guarantees that no element of $\{x_{ik}\}$ is repeated. The finiteness of the subsequence implies that the number of intermediate iterates must also be finite. This follows because a constraint is added to the working set (possibly with a zero step) for every intermediate iteration. Eventually, either a nonzero step will be taken, giving a strict decrease in φ, or enough constraints will be added to define a vertex (a trivial subspace minimizer).

4. Quadratic Programs in Standard Form

The inequality constraints of a quadratic program in standard form consist of only simple upper and lower bounds on the variables. Without loss of generality, we consider methods for the standard-form quadratic program:

$$\underset{x\in\mathbb{R}^n}{\text{minimize}}\;\; \varphi(x) = c^Tx + \tfrac{1}{2}x^THx \quad \text{subject to} \quad Ax = b, \quad x \ge 0. \tag{4.1}$$

This is an example of a mixed-constraint problem (1.1) with $D = I_n$ and $f = 0$. In this case, the working-set matrix D_w consists of rows of the identity matrix, and each working-set index i is associated with a variable x_i that is implicitly fixed at its current value. In this situation, as is customary for constraints in standard form, we refer to the working set as the *nonbasic set* $\mathcal{N}$, and denote its elements as $\{\nu_1, \nu_2, \ldots, \nu_{n_N}\}$ with $n_N = m_w$. The complementary set $\mathcal{B}$ of $n_B = n - n_N$ indices that are not in the working set is known as the *basic set*. The elements of the basic set are denoted by $\{\beta_1, \beta_2, \ldots, \beta_{n_B}\}$.

If P_N denotes the matrix of unit columns $\{\, e_i \,\}$ with $i \in \mathcal{N}$, then the working-set matrix A_w may be written as:

$$A_w = \begin{pmatrix} A \\ P_N^T \end{pmatrix}.$$

Similarly, if P_B is the matrix with unit columns $\{\, e_i \,\}$ with $i \in \mathcal{B}$, then $P = \begin{pmatrix} P_B & P_N \end{pmatrix}$ is a permutation matrix that permutes the columns of A_w as

$$A_w \begin{pmatrix} P_B & P_N \end{pmatrix} = A_wP = \begin{pmatrix} A \\ P_N^T \end{pmatrix} P = \begin{pmatrix} AP \\ P_N^TP \end{pmatrix} = \begin{pmatrix} A_B & A_N \\ & I_{n_N} \end{pmatrix},$$

where A_B and A_N are matrices with columns $\{\, a_{\beta_j} \,\}$ and $\{\, a_{\nu_j} \,\}$ respectively. If y is any n-vector, y_B (the *basic components of* y) denotes the n_B-vector whose j-th component is component β_j of y, and y_N (the *nonbasic components of* y) denotes the n_N-vector whose j-th component is component ν_j of y. We use the same convention for matrices, with the exception of I_B and I_N, which are reserved for the identity matrices of order n_B and n_N, respectively. With this notation, the effect of P on the Hessian and working-set matrix may be written as

$$P^THP = \begin{pmatrix} H_B & H_D \\ H_D^T & H_N \end{pmatrix}, \quad \text{and} \quad A_wP = \begin{pmatrix} A_B & A_N \\ & I_N \end{pmatrix}. \tag{4.2}$$

As in the generic mixed-constraint formulation, A_w must have full row-rank. This is equivalent to requiring that A_B has full row-rank since $\operatorname{rank}(A_w) = n_N + \operatorname{rank}(A_B)$.

For constraints in standard form, we say that x is a subspace minimizer with respect to the basic set $\mathcal{B}$ (or, equivalently, with respect to A_B). Similarly, a second-order-consistent working set is redefined as a *second-order-consistent basis.*

Result 4.1. (Subspace minimizer for standard form) *Let x be a feasible point with basic set $\mathcal{B}$. Let the columns of Z_B form a basis for the null-space of A_B.*

(a) *If x is a subspace stationary point with respect to A_w, then there exists a vector π such that $g_B = A_B^T\pi$, or equivalently, $Z_B^Tg_B = 0$.*

(b) *If $\mathcal{B}$ is a second-order-consistent basis, then $Z_B^T H_B Z_B$ is positive definite. Equivalently, the KKT matrix $K_B = \begin{pmatrix} H_B & A_B^T \\ A_B & \end{pmatrix}$ has inertia $(n_B, m, 0)$.* ∎

As in linear programming, the components of the vector $z = g(x) - A^T\pi$ are called the *reduced costs*. For constraints in standard form, the multipliers z_w associated inequality constraints in the working set are denoted by z_N. The components of z_N are the nonbasic components of the reduced-cost vector, i.e.,

$$z_N = (g(x) - A^T\pi)_{\mathcal{N}} = g_N - A_N^T\pi.$$

At a subspace stationary point, it holds that $g_B - A_B^T\pi = 0$, which implies that the basic components of the reduced costs z_B are zero.

The fundamental property of constraints in standard form is that the mixed-constraint method may be formulated so that the number of variables involved in the equality-constraint QP subproblem is reduced from n to n_B. By applying the permutation matrix P to the KKT equations (3.9a), we have

$$\left(\begin{array}{cc|cc} H_B & H_D & A_B^T & \\ H_D^T & H_N & A_N^T & I_N \\ \hline A_B & A_N & & \\ & I_N & & \end{array}\right) \begin{pmatrix} p_B \\ p_N \\ -q_\pi \\ -q_N \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \\ 0 \\ e_s \end{pmatrix}, \quad \text{where } p = P\begin{pmatrix} p_B \\ p_N \end{pmatrix} \text{ and } q = \begin{pmatrix} q_\pi \\ q_N \end{pmatrix}.$$

These equations imply that $p_N = e_s$ and p_B and q_π satisfy the reduced KKT system

$$\begin{pmatrix} H_B & A_B^T \\ A_B & 0 \end{pmatrix} \begin{pmatrix} p_B \\ -q_\pi \end{pmatrix} = \begin{pmatrix} -H_D p_N \\ -A_N p_N \end{pmatrix} = -\begin{pmatrix} (h_{\nu_s})_{\mathcal{B}} \\ a_{\nu_s} \end{pmatrix}. \tag{4.3}$$

In practice, p_N is defined implicitly and only the components of p_B and q_π are computed explicitly. Once p_B and q_π are known, the increment q_N for multipliers z_N associated with the constraints $p_N = e_s$ is given by $q_N = (Hp - A^T q_\pi)_{\mathcal{N}}$.

Similarly, the solution of the second KKT system (3.9b) can be computed from the KKT equation

$$\begin{pmatrix} H_B & A_B^T \\ A_B & \end{pmatrix} \begin{pmatrix} u_B \\ v_\pi \end{pmatrix} = \begin{pmatrix} e_r \\ 0 \end{pmatrix}, \tag{4.4}$$

with $u_N = 0$ and $v_N = -(Hu + A^T v_\pi)_{\mathcal{N}}$, where $u = P\begin{pmatrix} u_B \\ u_N \end{pmatrix}$ and $v = \begin{pmatrix} v_\pi \\ v_N \end{pmatrix}$.

The KKT equations (4.3) and (4.4) allow the mixed constraint algorithm to be formulated in terms of the basic variables only, which implies that the algorithm is driven by variables entering or leaving the basic set rather than constraints entering or leaving the working set. With this interpretation, changes to the KKT matrix are based on column-changes to A_B instead of row-changes to D_w.

For completeness we summarize Results 3.2–3.5 in terms of the quantities associated with constraints in standard form (an explicit proof of each result is given by Wong [55]).

Result 4.2. *Let x be a subspace minimizer with respect to the basic set $\mathcal{B}$, with $[z_N]_s < 0$. Let $\bar{x}$ be the point such that $\bar{x}_N = x_N + \alpha e_s$ and $\bar{x}_B = x_B + \alpha p_B$, where p_B is defined as in (4.3).*

(1) *The step to the minimizer of $\varphi(x + \alpha p)$ is $\alpha_* = -z_{\nu_s}/[q_N]_s$. If α_* is bounded and $\alpha = \alpha_*$, then $\bar{x}$ is a subspace minimizer with respect to the basic set $\bar{\mathcal{B}} = \mathcal{B} + \{\nu_s\}$.*

(2) *The largest feasible step is defined using the minimum ratio test:*

$$\alpha_F = \min \gamma_i, \quad \textit{where} \quad \gamma_i = \begin{cases} \dfrac{[x_B]_i}{-[p_B]_i} & \textit{if } [p_B]_i < 0, \\ +\infty & \textit{otherwise.} \end{cases}$$

Suppose $\alpha = \alpha_F$ *and* $[x_B + \alpha p_B]_{\beta_r} = 0$ *and let* u_B *and* v_π *be defined by* (4.4).

(a) e_r *and the columns of* A_B^T *are linearly independent if and only if* $u_B \neq 0$.

(b) $[v_N]_s = [p_B]_r < 0$ *and* $[u_B]_r \geq 0$, *with* $[u_B]_r > 0$ *if* $u_B \neq 0$.

(c) *If* e_r *and the columns of* A_B^T *are linearly independent, then* $\bar{x}$ *is a subspace minimizer with respect to* $\bar{\mathcal{B}} = \mathcal{B} - \{\beta_r\}$. *Moreover,* $g_{\bar{B}}(\bar{x}) = A_{\bar{B}}^T \bar{\pi}$ *and* $g_{\bar{N}}(\bar{x}) = A_{\bar{N}}^T \bar{\pi} + \bar{z}_N$, *where* $\bar{\pi} = \pi + \alpha q_\pi$ *and* $\bar{z}_N$ *is formed by appending a zero component to the vector* $z_N + \alpha q_N$.

(d) *If* e_r *and the columns of* A_B^T *are linearly dependent, define* $\sigma = [z_N + \alpha q_N]_s / [v_N]_s$. *Then* $\bar{x}$ *is a subspace minimizer with respect to* $\bar{\mathcal{B}} = \mathcal{B} - \{\beta_r\} + \{\nu_s\}$ *with* $g_{\bar{B}}(\bar{x}) = A_{\bar{B}}^T \bar{\pi}$ *and* $g_{\bar{N}}(\bar{x}) = A_{\bar{N}}^T \bar{\pi} + \bar{z}_N$, *where* $\bar{\pi} = \pi + \alpha q_\pi - \sigma v_\pi$ *with* $\sigma > 0$, *and* $\bar{z}_N$ *is formed by appending* σ *to* $z_N + \alpha q_N - \sigma v_N$. ∎

As in the generic mixed-constraint method, the direction p_B and multiplier q_π may be updated in the linearly independent case.

Result 4.3. *Let* x *be a subspace minimizer with respect to* $\mathcal{B}$. *Assume the vectors* p_B, q_π, u_B *and* v_π *are defined by* (4.3) *and* (4.4). *Let* β_r *be the index of a linearly independent blocking variable at* $\bar{x}$, *where* $\bar{x}_N = x_N + \alpha e_s$ *and* $\bar{x}_B = x_B + \alpha p_B$. *Let* $\rho = -[p_B]_r / [u_B]_r$, *and consider the vectors* $\bar{p}_B$ *and* $\bar{q}_\pi$, *where* $\bar{p}_B$ *is the vector* $p_B + \rho u_B$ *with the* r*-th component omitted, and* $\bar{q}_\pi = q_\pi - \rho v_\pi$. *Then* $\bar{p}_B$ *and* $\bar{q}_\pi$ *are well-defined and satisfy the KKT equations for the basic set* $\mathcal{B} - \{\beta_r\}$. ∎

Linear programming. If the problem is a linear program (i.e., $H = 0$), then the basic set $\mathcal{B}$ must be chosen so that A_B is nonsingular (i.e., it is square with rank m). In this case, we show that Algorithm 4.1 simplifies to a variant of the primal simplex method in which the π-values and reduced costs are updated by a simple recurrence relation.

When $H = 0$, the equations (4.3) reduce to $A_B p_B = -a_{\nu_s}$ and $A_B^T q_\pi = 0$, with $p_N = e_s$ and $q_N = -A_N^T q_\pi$. As A_B is nonsingular, both q_π and q_N are zero, and the directions p_B and p_N are equivalent to those defined by the simplex method. For the singularity test (4.4), the basic and nonbasic components of u satisfy $A_B u_B = 0$ and $u_N = 0$. Similarly, $v_N = -A_N^T v_\pi$, where $A_B^T v_\pi = e_r$, As A_B is nonsingular, $u_B = 0$ and the linearly dependent case always applies. This implies that the r-th basic and the s-th nonbasic variables are always swapped, as in the primal simplex method.

As q is zero, the updates to the multiplier vectors π and z_N defined by part 2(d) of Result 4.2 depend only on the vectors v_π and v_N, and the scalar $\sigma = [z_N]_s / [p_B]_r$. The resulting updates to the multipliers are:

$$\pi \leftarrow \pi - \sigma v_\pi, \quad \text{and} \quad z_N \leftarrow \begin{pmatrix} z_N - \sigma v_N \\ \sigma \end{pmatrix},$$

which are the established multiplier updates associated with the simplex method (see Gill [23] and Tomlin [54]). It follows that the simplex method is a method for which every subspace minimizer is standard.

Summary and discussion. Algorithm 4.1 summarizes the method for general QPs in standard form. (The relation in part 2(b) of Result 4.2 is used to simplify the computation of $[v_N]_s$.) Given an arbitrary feasible point x_0, and a second-order-consistent basis $\mathcal{B}_0$, Algorithm 4.1 generates a sequence of primal-dual iterates $\{(x_k, y_k)\}$ and associated basic sets $\mathcal{B}_k$ such that

$$\begin{pmatrix} x_{k+1} \\ y_{k+1} \end{pmatrix} = \begin{pmatrix} x_k \\ y_k \end{pmatrix} + \alpha_k \begin{pmatrix} p_k \\ q_k \end{pmatrix},$$

where p_k and q_k are either computed directly by solving (4.3), or are updated from previous values using the solution of (4.4).

The algorithm starts by attempting to minimize the objective with respect to the basic variables in $\mathcal{B}_0$. If the minimizer is infeasible, the quadratic objective is minimized over a sequence of nested basic sets until enough blocking variables are fixed on their bounds to define a subspace minimizer (e.g., at a vertex, which is trivially a subspace minimizer). Once the first subspace minimizer is found, the iterates occur in groups of iterates that start and finish at a standard subspace minimizer. Each group starts with the identification of a nonbasic variable x_{ν_s} with a negative reduced cost z_{ν_s}. In the group of subsequent iterations, the reduced cost z_{ν_s} is driven to zero. During each of these *intermediate* iterations, the nonbasic variable x_{ν_s} is allowed to move away from its bound, and a blocking basic variable may be made nonbasic to maintain feasibility. Once z_{ν_s} reaches zero, the associated nonbasic variable x_{ν_s} is moved into the basic set. Figure 1 depicts a sequence of intermediate iterations starting at a subspace minimizer with respect to $\mathcal{B}_0$. The figure illustrates the two ways in which the algorithm arrives at a point with a zero value of z_{ν_s} (i.e., at a subspace minimizer). In case (A), x_{j+1} is the result of an unconstrained step along p_j. In case (B), the removal of the blocking variable from the basic set would give a rank-deficient basis and the blocking index must be swapped with the nonbasic index ν_s (see part (d) of Result 4.2).

For each intermediate iteration, the definition of the optimal step α_* involves the curvature $[q_N]_s = p^T H p$, which represents the rate of change of the reduced cost z_{ν_s} in the direction p. This curvature increases monotonically over the sequence of intermediate iterates, which implies that the curvature becomes "less negative" as blocking basic variables are made nonbasic. For a convex QP, it holds that $p^T H p \geq 0$, which implies that only the first direction associated with a group of consecutive iterates can be a direction of zero curvature. Figure 2 depicts three examples of the behavior of the nonbinding multiplier $z_{\nu_s}(\sigma)$ as x varies along the piecewise linear path $x(\sigma)$ joining the sequence of intermediate iterates. The nonbinding multiplier $z_{\nu_s}(\sigma)$ is a continuous, piecewise linear function, with a discontinuous derivative at any point where a blocking variable is made nonbasic. The value of $z_{\nu_s}(0)$ is z_{ν_s}, the (negative) reduced cost at the first standard subspace minimizer. The slope of each segment is given by the value of the curvature $\theta_j = p_j^T H p_j$ along the direction of each segment of the path $x(\sigma)$. As the iterations proceed, the nonbinding multiplier is driven to zero, and the intermediate iterations terminate at the point where $z_{\nu_s}(\sigma) = 0$. As a variable moves from basic to nonbasic along the piecewise linear path, the slope of the z-segment becomes more positive. In the left-most figure, the curvature starts at a positive value, which always holds for a strictly convex problem, and is typical for a convex problem with a nonzero H. In the right-most figure, the curvature starts at zero, which is possible for a convex problem with a singular H, and is always the case for a linear program. If the problem is unbounded, then $z_{\nu_s}(\sigma)$ remains at the fixed negative value $z_{\nu_s}(0)$ for all $\sigma \geq 0$. In the lower figure, the initial curvature is negative, and p is a direction of negative curvature. This situation may occur for a nonconvex problem. In this case $z_{\nu_s}(\sigma)$ may remain negative for a number of intermediate iterations. If the problem is unbounded, then $z_{\nu_s}(\sigma)$ is unbounded below for increasing σ.

	x_0	$\longrightarrow$	$\cdots$	x_{j-1}	$\longrightarrow$	x_j	$\longrightarrow$	x_{j+1}
(A)	$\mathcal{B}_0$	*move, delete* $\longrightarrow$	$\cdots$	$\mathcal{B}_{j-1}$	*move, delete* $\longrightarrow$	$\mathcal{B}_j$	*move and add* ν_s $\longrightarrow$	$\mathcal{B}_{j+1}$
(B)	$\mathcal{B}_0$	*move, delete* $\longrightarrow$	$\cdots$	$\mathcal{B}_{j-1}$	*move, delete* $\longrightarrow$	$\mathcal{B}_j$	*move and swap* $\longrightarrow$	$\mathcal{B}_{j+1}$

Figure 1: This figure illustrates the structure of a typical sequence of iterations that follow the identification of a nonoptimal reduced cost. Each sequence consists of $j+2$ iterates that begin and end at the standard subspace minimizers x_0 and x_{j+1}. The j ($j \geq 0$) intermediate iterates are nonstandard subspace minimizers. In (A), x_{j+1} is reached by taking an unconstrained step along p_j. In (B), the removal of the blocking variable from the basic set would give a rank-deficient basis and the index of the blocking variable is swapped with the index of the nonbinding nonbasic variable. The point x_{j+1} is the first standard minimizer for the next sequence.

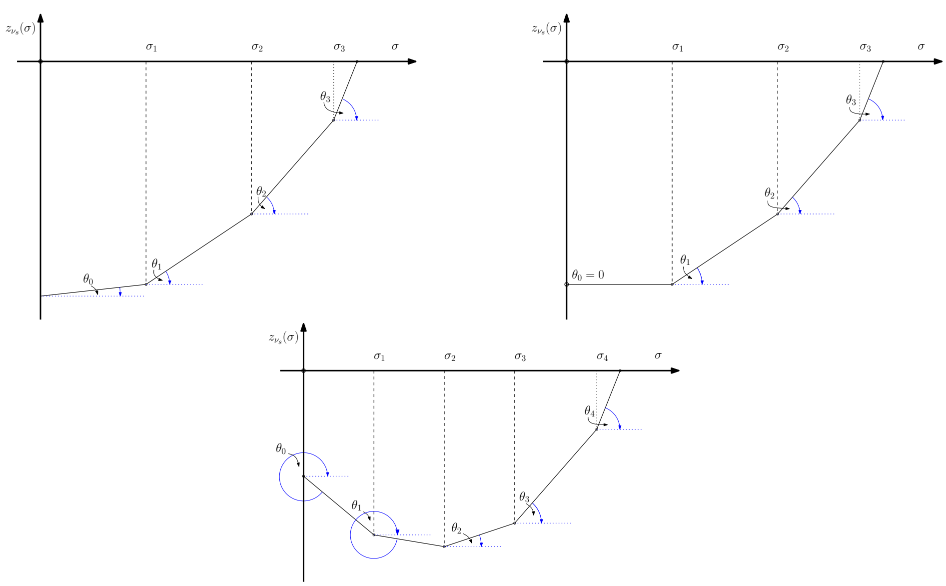

Figure 2: Three examples of the behavior of the nonbinding multiplier $z_{\nu_s}(\sigma)$ as x varies along the piecewise linear path $x(\sigma)$ joining the sequence of intermediate iterates. The function $z_{\nu_s}(\sigma)$ is piecewise linear with $z_{\nu_s}(0) < 0$, and slopes $\theta_j = p_j^T H p_j$ that increase monotonically as blocking variables are made nonbasic. As the iterations proceed, the nonbinding multiplier is driven to zero, and the intermediate iterations terminate at the point where $z_{\nu_s}(\sigma) = 0$. The left-most figure depicts a convex problem for which the curvature starts at a positive value. The right-most figure depicts a convex problem for which the curvature starts at zero. The lower figure depicts a nonconvex problem for which the curvature starts at a negative value.

Algorithm 4.1. [Method for a general QP in standard form]

Find x_0 such that $Ax_0 = b$ and $x_0 \geq 0$;
$[x, \pi, \mathcal{B}, \mathcal{N}] = \texttt{subspaceMin}(x_0)$; [find a subspace minimizer]
$g = c + Hx$; $z = g - A^T\pi$;
$\nu_s = \text{argmin}_i\{z_i\}$; [identify the least-optimal multiplier]
while $z_{\nu_s} < 0$ **do** [drive z_{ν_s} to zero]
  Solve $\begin{pmatrix} H_B & A_B^T \\ A_B & \end{pmatrix}\begin{pmatrix} p_B \\ -q_\pi \end{pmatrix} = -\begin{pmatrix} (h_{\nu_s})_{\mathcal{B}} \\ a_{\nu_s} \end{pmatrix}$; $p_N = e_s$;
  repeat
    $p = P\begin{pmatrix} p_B \\ p_N \end{pmatrix}$; $q_N = (Hp - A^T q_\pi)_{\mathcal{N}}$;
    $\alpha_F = \texttt{minRatioTest}(x_B, p_B)$; [compute the largest step to a blocking variable]
    if $[q_N]_s > 0$ **then** $\alpha_* = -z_{\nu_s}/[q_N]_s$ **else** $\alpha_* = +\infty$; [compute the optimal step]
    $\alpha = \min\{\alpha_*, \alpha_F\}$;
    if $\alpha = +\infty$ **then stop**; [unbounded solution]
    $x \leftarrow x + \alpha p$; $g \leftarrow g + \alpha H p$;
    $\pi \leftarrow \pi + \alpha q_\pi$; $z = g - A^T\pi$;
    if $\alpha_F < \alpha_*$ **then**
      Find the index r of a blocking variable;
      Solve $\begin{pmatrix} H_B & A_B^T \\ A_B & \end{pmatrix}\begin{pmatrix} u_B \\ v_\pi \end{pmatrix} = \begin{pmatrix} e_r \\ 0 \end{pmatrix}$;
      if $u_B = 0$ **then**
        $\sigma = z_{\nu_s}/[p_B]_r$; $\pi \leftarrow \pi - \sigma v_\pi$;
        $z = g - A^T\pi$; [implies $z_{\nu_s} = 0$]
      else
        $\rho = -[p_B]_r/[u_B]_r$;
        $p_B \leftarrow p_B + \rho u_B$; $q_\pi \leftarrow q_\pi - \rho v_\pi$;
      end
      $\mathcal{B} \leftarrow \mathcal{B} - \{\beta_r\}$; $\mathcal{N} \leftarrow \mathcal{N} + \{\beta_r\}$; [make the blocking variable β_r nonbasic]
    end;
  until $z_{\nu_s} = 0$;
  $\mathcal{B} \leftarrow \mathcal{B} + \{\nu_s\}$; $\mathcal{N} \leftarrow \mathcal{N} - \{\nu_s\}$; [make variable ν_s basic]
  $\nu_s = \text{argmin}_i\{z_i\}$;
  $k \leftarrow k + 1$;
end do

5. Solving the KKT Systems

At each iteration of the primal methods discussed in Sections 4, it is necessary to solve one or two systems of the form

$$\begin{pmatrix} H_B & A_B^T \\ A_B & \end{pmatrix} \begin{pmatrix} y \\ w \end{pmatrix} = \begin{pmatrix} h \\ f \end{pmatrix}, \tag{5.1}$$

where h and f are given by right-hand sides of the equations (4.3) or (4.4). Two alternative approaches for solving (5.1) are described. The first involves the symmetric transformation of the KKT system into three smaller systems, one of which involves the explicit reduced Hessian matrix. The second approach uses a symmetric indefinite factorization of a fixed KKT matrix in conjunction with the factorization of a smaller matrix that is updated at each iteration.

5.1. Variable reduction

The variable-reduction method involves transforming the equations (5.1) to block-triangular form using the nonsingular block-diagonal matrix $\operatorname{diag}(Q, I_m)$. Consider a column permutation P such that

$$AP = \begin{pmatrix} B & S & N \end{pmatrix},$$

with B an $m \times m$ nonsingular matrix and S an $m \times n_S$ matrix with $n_S = n_B - m$. The matrix P is a version of the permutation $P = \begin{pmatrix} P_B & P_N \end{pmatrix}$ of (4.2) that also arranges the columns of A_B in the form $A_B = \begin{pmatrix} B & S \end{pmatrix}$. The n_S variables associated with S are called the *superbasic* variables. Given P, consider the nonsingular $n \times n$ matrix Q such that

$$Q = P \begin{pmatrix} -B^{-1}S & I_m & 0 \\ I_{n_S} & 0 & 0 \\ 0 & 0 & I_N \end{pmatrix}.$$

The columns of Q may be partitioned so that $Q = \begin{pmatrix} Z & Y & W \end{pmatrix}$, where

$$Z = P \begin{pmatrix} -B^{-1}S \\ I_{n_S} \\ 0 \end{pmatrix}, \quad Y = P \begin{pmatrix} I_m \\ 0 \\ 0 \end{pmatrix} \quad \text{and} \quad W = P \begin{pmatrix} 0 \\ 0 \\ I_N \end{pmatrix}.$$

The columns of the $n \times n_S$ matrix Z form a basis for the null-space of A_w, with

$$A_w Q = \begin{pmatrix} A \\ P_N^T \end{pmatrix} Q = \begin{pmatrix} 0 & B & N \\ 0 & 0 & I_N \end{pmatrix}.$$

Suppose that we wish to solve a generic KKT system

$$\begin{pmatrix} H & A^T & P_N \\ A & & \\ P_N^T & & \end{pmatrix} \begin{pmatrix} y \\ w_1 \\ w_2 \end{pmatrix} = \begin{pmatrix} h \\ f_1 \\ f_2 \end{pmatrix}.$$

Then the vector y may be computed as $y = Y y_Y + Z y_Z + W y_W$, where y_Y, y_Z, y_W and w are defined using the equations

$$\begin{pmatrix} Z^THZ & Z^THY & Z^THW & & \\ Y^THZ & Y^THY & Y^THW & B^T & \\ W^THZ & W^THY & W^THW & N^T & I_N \\ & B & N & & \\ & & I_N & & \end{pmatrix} \begin{pmatrix} y_Z \\ y_Y \\ y_W \\ w_1 \\ w_2 \end{pmatrix} = \begin{pmatrix} h_Z \\ h_Y \\ h_W \\ f_1 \\ f_2 \end{pmatrix}, \tag{5.2}$$

with $h_Z = Z^T h$, $h_Y = Y^T h$, and $h_W = W^T h$. This leads to

$$\begin{aligned} y_W &= f_2, \\ B y_Y &= f_1 - N f_2, & y_R &= Y y_Y + W y_W, \\ Z^T H Z y_Z &= Z^T(h - H y_R), & y_T &= Z y_Z, & y &= y_R + y_T, \\ B^T w_1 &= Y^T(h - Hy), & w_2 &= W^T(h - Hy) - N^T w_1. \end{aligned}$$

The equations simplify considerably for the KKT systems (3.9a) and (3.9b). In the case of (3.9a), the equations are:

$$\begin{aligned} B p_Y &= -a_{\nu_s}, & p_R &= P \begin{pmatrix} p_Y \\ 0 \\ e_s \end{pmatrix}, \\ Z^T H Z p_Z &= -Z^T H p_R, & p_T &= Z p_Z, & p &= p_R + p_T, \\ B^T q_\pi &= \ (Hp)_{\mathcal{B}}, & q_z &= (Hp - A^T q_\pi)_{\mathcal{N}}. \end{aligned} \tag{5.3}$$

Similarly for (3.9b), it holds that $u_Y = 0$, $u_R = 0$, and

$$\begin{aligned} Z^T H Z u_Z &= Z^T e_{\beta_r}, & u &= Z u_Z, \\ B^T v_\pi &= (e_{\beta_r} - Hu)_{\mathcal{B}}, & v_z &= -(Hu + A^T v_\pi)_{\mathcal{N}}. \end{aligned} \tag{5.4}$$

These equations allow us to specialize Part 2(a) of Result 4.2, which gives the conditions for the linear independence of the rows of the new A_B.

Result 5.1. *Let x be a subspace minimizer with respect to the basic set $\mathcal{B}$. Assume that p and q are defined by (4.3), and that x_{β_r} is the variable selected to be nonbasic at the next iterate. Let the vectors u_B and v_π be defined by (4.4).*

(a) *If x_{β_r} is superbasic, then e_r and the rows of A_B are linearly independent (i.e., the matrix obtained by removing the rth column of A_B has rank m).*

(b) *If x_{β_r} is not superbasic, then e_r is linearly independent of the rows of A_B if and only if $S^T z \neq 0$, where z is the solution of $B^T z = e_r$.*

Proof. From (5.4), $u = Z u_Z$, which implies that u_B is nonzero if and only if u_Z is nonzero. Similarly, the nonsingularity of $Z^T H Z$ implies that u_Z is nonzero if and only if $Z^T e_{\beta_r}$ is nonzero. Now

$$Z^T e_{\beta_r} = \begin{pmatrix} -S^T B^{-T} & I_{n_S} & 0 \end{pmatrix} e_r.$$

If x_{β_r} is superbasic, then $r > m$ and $Z^T e_{\beta_r} = e_{r-m} \neq 0$ and u_Z is nonzero. If x_{β_r} is not superbasic, then $r \leq m$, and

$$Z^T e_{\beta_r} = -S^T B^{-T} e_r = -S^T z,$$

where z is the solution of $B^T z = e_r$. ∎

The equations (5.3) and (5.4) may be solved using a Cholesky factorization of $Z^T H Z$ and an LU factorization of B. The factors of B allow efficient calculation of matrix-vector products $Z^T v$ or $Z v$ without the need to form the inverse of B.

5.2. Fixed-factorization updates

When A_B and H_B are large and sparse, there are many reliable and efficient sparse-matrix factorization packages for solving a symmetric indefinite system of the form (5.1). Some prominent software packages include `MA27` (Duff and Reid [18]), `HSL_MA57` (Duff [16]), `MUMPS` (Amestoy et al. [1]), `PARDISO` (Schenk and Gärtner [53]), and `SPOOLES` (Ashcraft and Grimes [2]). However, in a QP algorithm, a sequence of related systems must be solved in which the KKT matrix changes by a single row and column. In this situation, instead of factoring the matrix in (5.1) directly, the first K_0 may be "bordered" in a way that reflects the changes to the basic and nonbasic sets during a set of k subsequent iterations. The solution of (5.1) is then found by using a *fixed* factorization of K_0, and a factorization of a smaller matrix of (at most) order k (see Bisschop and Meeraus [4], and Gill et al. [32]). Although K_0 is symmetric, the matrix may be factored by any symmetric or unsymmetric linear solver, allowing a variety of black-box linear solvers to be incorporated into the algorithm.

Let $\mathcal{B}_0$ and $\mathcal{N}_0$ denote the initial basic and nonbasic sets that define the KKT system (5.1). There are four cases to consider:

(1) a nonbasic variable moves to the basic set and is not in $\mathcal{B}_0$,

(2) a basic variable in $\mathcal{B}_0$ becomes nonbasic,

(3) a basic variable not in $\mathcal{B}_0$ becomes nonbasic, and

(4) a nonbasic variable moves to the basic set and is in $\mathcal{B}_0$.

For case (1), let ν_s be the nonbasic variable that has become basic. The next KKT matrix can be written as

$$\left(\begin{array}{cc|c} H_B & A_B^T & (h_{\nu_s})_{\mathcal{B}_0} \\ A_B & 0 & a_{\nu_s} \\ \hline (h_{\nu_s})_{\mathcal{B}_0}^T & a_{\nu_s}^T & h_{\nu_s,\nu_s} \end{array}\right).$$

Suppose that at the next stage, another nonbasic variable ν_r becomes basic. The KKT matrix is augmented in a similar fashion, i.e.,

$$\left(\begin{array}{ccc|c} H_B & A_B^T & (h_{\nu_s})_{\mathcal{B}_0} & (h_{\nu_r})_{\mathcal{B}_0} \\ A_B & 0 & a_{\nu_s} & a_{\nu_r} \\ (h_{\nu_s})_{\mathcal{B}_0}^T & a_{\nu_s}^T & h_{\nu_s,\nu_s} & h_{\nu_s,\nu_r} \\ \hline (h_{\nu_r})_{\mathcal{B}_0}^T & a_{\nu_r}^T & h_{\nu_r,\nu_s} & h_{\nu_r,\nu_r} \end{array}\right).$$

Now consider case 2 and let $\beta_r \in \mathcal{B}_0$ become nonbasic. The change to the basic set is reflected in the new KKT matrix

$$\left(\begin{array}{cccc|c} H_B & A_B^T & (h_{\nu_s})_{\mathcal{B}_0} & (h_{\nu_r})_{\mathcal{B}_0} & e_r \\ A_B & 0 & a_{\nu_s} & a_{\nu_r} & 0 \\ (h_{\nu_s})_{\mathcal{B}_0}^T & a_{\nu_s}^T & h_{\nu_s,\nu_s} & h_{\nu_s,\nu_r} & 0 \\ (h_{\nu_r})_{\mathcal{B}_0}^T & a_{\nu_r}^T & h_{\nu_r,\nu_s} & h_{\nu_r,\nu_r} & 0 \\ \hline e_r^T & 0 & 0 & 0 & 0 \end{array}\right).$$

The unit row and column augmenting the matrix has the effect of zeroing out the components corresponding to the removed basic variable.

In case (3), the basic variable must have been added to the basic set at a previous stage as in case (1). Thus, removing it from the basic set can be done by removing the row and column in the augmented part of the KKT matrix corresponding to its addition to the basic set. For example, if ν_s is the basic to be removed, then the new KKT matrix is given by

$$\begin{pmatrix} H_B & A_B^T & (h_{\nu_r})_{\mathcal{B}_0} & e_r \\ A_B & 0 & a_{\nu_r} & 0 \\ (h_{\nu_r})_{\mathcal{B}_0}^T & a_{\nu_r}^T & h_{\nu_r,\nu_r} & 0 \\ e_r^T & 0 & 0 & 0 \end{pmatrix}.$$

For case (4), a nonbasic variable in $\mathcal{B}_0$ implies that at some previous stage, the variable was removed from $\mathcal{B}_0$ as in case (2). The new KKT matrix can be formed by removing the unit row and column in the augmented part of the KKT matrix corresponding to the removal the variable from the basic set. In this example, the new KKT matrix becomes

$$\begin{pmatrix} H_B & A_B^T & (h_{\nu_r})_{\mathcal{B}_0} \\ A_B & 0 & a_{\nu_r} \\ (h_{\nu_r})_{\mathcal{B}_0}^T & a_{\nu_r}^T & h_{\nu_r,\nu_r} \end{pmatrix}.$$

After k iterations, the KKT system is maintained as a symmetric augmented system of the form

$$\begin{pmatrix} K & V \\ V^T & D \end{pmatrix} \begin{pmatrix} r \\ \eta \end{pmatrix} = \begin{pmatrix} b \\ f \end{pmatrix} \quad \text{with} \quad K = \begin{pmatrix} H_B & A_B^T \\ A_B & \end{pmatrix}, \tag{5.5}$$

where D is of dimension at most $2k$.

5.2.1. Schur complement and block LU methods

Although the augmented system (in general) increases in dimension by one at every iteration, the 1×1 block K is fixed and defined by the initial set of basic variables. The *Schur complement method* assumes that factorizations for K and the *Schur complement* $C = D - V^T K^{-1} V$ exist. Then the solution of (5.5) can be determined by solving the equations

$$Kt = b, \quad C\eta = f - V^T t, \quad Kr = b - V\eta.$$

The work required is dominated by two solves with the fixed matrix K and one solve with the Schur complement C. If the number of changes to the basic set is small enough, dense factors of C may be maintained.

The Schur complement method can be extended to a *block LU method* by storing the augmented matrix in block factors

$$\begin{pmatrix} K & V \\ V^T & D \end{pmatrix} = \begin{pmatrix} L & \\ Z^T & I \end{pmatrix} \begin{pmatrix} U & Y \\ & C \end{pmatrix}, \tag{5.6}$$

where $K = LU$, $LY = V$, $U^T Z = V$, and $C = D - Z^T Y$ is the Schur-complement matrix.

The solution of (5.5) can be computed by forming the block factors and by solving the equations

$$Lt = b, \quad C\eta = f - Z^T t, \quad Ur = t - Y\eta.$$

This method requires a solve with L and U each, one multiply with Y and Z^T, and one solve with the Schur complement C. For more details, see Gill et al. [29], Eldersveld and Saunders [19], and Huynh [45].

As the iterations of the QP algorithm proceed, the size of C increases and the work required to solve with C increases. It may be necessary to restart the process by discarding the existing factors and re-forming K based on the current set of basic variables.

5.2.2. Updating the block LU factors

Suppose the current KKT matrix is bordered by the vectors v and w, and the scalar σ

$$\left(\begin{array}{cc|c} K & V & v \\ V^T & D & w \\ \hline v^T & w^T & \sigma \end{array}\right).$$

The block LU factors Y and Z, and the Schur complement C are updated every time the system is bordered. The number of columns in matrices Y and Z and the dimension of the Schur complement increase by one. The updates y, z, c and d are defined by the equations

$$\begin{aligned} Ly &= v, & U^Tz &= v, \\ c = w - Z^Ty &= w - Y^Tz, & d &= \sigma - z^Ty, \end{aligned}$$

so that the new block LU factors satisfy

$$\left(\begin{array}{c|cc} K & V & v \\ \hline V^T & D & w \\ v^T & w^T & \sigma \end{array}\right) = \left(\begin{array}{c|cc} L & & \\ \hline Z^T & I & \\ z^T & & 1 \end{array}\right) \left(\begin{array}{c|cc} U & Y & y \\ \hline & C & c \\ & c^T & d \end{array}\right).$$

6. Finding a Subspace Minimizer

The method described in Section 4 have the property that if the initial iterate x_0 is a subspace minimizer, then all subsequent iterates are subspace minimizers (see Result 4.2). Methods for finding an initial subspace minimizer utilize an initial estimate x_I of the solution together with matrices A_B and A_N associated with an estimate of the optimal basic and nonbasic partitions of A. These estimates are often available from the known solution of a related QP—e.g., from the solution of the previous QP subproblem in the SQP context. The initial point x_I may or may not be feasible, and the associated matrix A_B may or may not have rank m.

The definition of a second-order-consistent basis requires that the matrix A_B has rank m, and it is necessary to identify a set of linearly independent basic columns of A. One algorithm for doing this has been proposed by Gill, Murray and Saunders [27], who use a sparse LU factorization of A_B^T to identify a square nonsingular subset of the columns of A_B. If necessary, a "basis repair" scheme is used to define additional unit columns that make A_B have full rank. The nonsingular matrix B obtained as a by-product of this process may be expressed in terms of A using a column permutation P such that

$$AP = \begin{pmatrix} A_B & A_N \end{pmatrix} = \begin{pmatrix} B & S & A_N \end{pmatrix}. \tag{6.1}$$

Given x_I, a point x_0 satisfying $Ax = b$ may be computed as

$$x_0 = x_I + P\begin{pmatrix} p_Y \\ 0 \\ 0 \end{pmatrix}, \quad \text{where} \quad Bp_Y = -(Ax_I - b).$$

If the matrix

$$K_B = \begin{pmatrix} H_B & A_B^T \\ A_B & \end{pmatrix} \tag{6.2}$$

has n_B positive eigenvalues and m negative eigenvalues, then the inertia of K_B is correct and x_0 is used as the initial point for a sequence of Newton-type iterations in which $\varphi(x)$

is minimized with the nonbasic components of x fixed at their current values. Consider the equations

$$\begin{pmatrix} H_B & A_B^T \\ A_B & \end{pmatrix} \begin{pmatrix} p_B \\ -\pi \end{pmatrix} = -\begin{pmatrix} g_B \\ 0 \end{pmatrix}.$$

If p_B is zero, x is a subspace stationary point (with respect to A_B) at which K_B has correct inertia and we are done. If p_B is nonzero, two situations are possible.

If $x_B + p_B$ is infeasible, then feasibility is retained by determining the maximum nonnegative step $\alpha < 1$ such that $x_B + \alpha p_B$ is feasible. A variable on its bound at $x_B + \alpha p_B$ is then removed from the basic set and the iteration is repeated. The removal of a basic variable cannot increase the number of negative eigenvalues of K_B and a subspace minimizer must be determined in a finite number of steps.

If $x_B + p_B$ is feasible, then p_B is the step to the minimizer of $\varphi(x)$ with respect to the basic variables and it must hold that $x_B + p_B$ is a subspace minimizer.

A KKT matrix with incorrect inertia has too many negative or zero eigenvalues. In this case, an appropriate K_B may be obtained by imposing temporary constraints that are deleted during the course of subsequent iterations. For example, if $n - m$ variables are temporarily fixed at their current values, then A_B is a square nonsingular matrix and K_B necessarily has exactly m negative eigenvalues. The form of the temporary constraints depends on the method used to solve the reduced KKT equations (5.1).

6.1. Variable-reduction method

In the variable reduction method a dense Cholesky factor of the reduced Hessian $Z^T HZ$ is updated to reflect changes in the basic set. (see Section 5.1). At the initial x_0 a partial Cholesky factorization with interchanges is used to find an upper-triangular matrix R that is the factor of the largest positive-definite leading submatrix of $Z^T HZ$. The use of interchanges tends to maximize the dimension of R. Let Z_R denote the columns of Z corresponding to R, and let Z be partitioned as $Z = \begin{pmatrix} Z_R & Z_A \end{pmatrix}$. A nonbasic set for which Z_R defines an appropriate null space can be obtained by fixing the variables corresponding to the columns of Z_A at their current values. As described above, minimization of $\varphi(x)$ then proceeds within the subspace defined by Z_R. If a variable is removed from the basic set, a row and column is removed from the reduced Hessian and an appropriate update is made to the Cholesky factor.

6.2. Fixed factorization updates

If fixed factorization updates to the KKT matrix are being used, the procedure for finding a second-order-consistent basis is given as follows.

1. Factor the reduced KKT matrix (6.2) system in the form $K_B = LDL^T$, where L is unit lower-triangular and D is block diagonal with 1×1 and 2×2 blocks. If the inertia of K_B is correct, then we are done.

2. If the inertia is incorrect, factor

   $$H_A = H_B + \rho A_B^T A_B = L_A D_A L_A^T,$$

   where ρ is a modest positive penalty parameter. As the inertia of K_B is not correct, D_A will have some negative eigenvalues for all positive ρ.

   The factorization of H_A may be written in the form

   $$H_A = L_A U \Lambda U^T L_A^T = V \Lambda V^T,$$

where $U\Lambda U^T$ is the spectral decomposition of D_A. The block diagonal structure of D_A implies that U is a block-diagonal orthonormal matrix. The inertia of Λ is the same as the inertia of H_A, and there exists a positive semidefinite diagonal matrix E such that $\Lambda + E$ is positive definite. If $\bar{H}_A$ is the positive-definite matrix $V(\Lambda + E)V^T$, then

$$\bar{H}_A = H_A + VEV^T = H_A + \sum_{e_{jj}>0} e_{jj} v_j v_j^T.$$

Suppose that H_A has r nonpositive eigenvalues. Define V_B as the $r \times n_B$ matrix consisting of the columns of V associated with the positive components of E. The augmented KKT matrix

$$\begin{pmatrix} H_B & A_B^T & V_B \\ A_B & 0 & 0 \\ V_B^T & 0 & 0 \end{pmatrix}$$

has exactly $m + r$ negative eigenvalues and hence has correct inertia.

The minimization of $\varphi(x)$ proceeds subject to the original constraints and the (general) temporary constraints $V_B^T x_B = 0$.

The efficiency of this scheme will depend on the number of surplus negative and zero eigenvalues in H_A. In practice, if the number of negative eigenvalues exceeds a preassigned threshold, then a temporary vertex is defined by fixing the variables associated with the columns of S in (6.1) (see Section 7).

7. Numerical Results

7.1. Implementation

The package SQIC is a Fortran 2008 implementation of the general quadratic programming method discussed in Section 4. SQIC is designed to solve large-scale problems of the form

$$\underset{x,s}{\text{minimize}} \;\; \varphi(x) = c^T x + \tfrac{1}{2} x^T H x \quad \text{subject to} \quad Ax - s = 0, \quad l \le \begin{pmatrix} x \\ s \end{pmatrix} \le u,$$

where s is the vector of slack variables, l and u are constant lower and upper bounds, and A and H are sparse matrices of dimension $m \times n$ and $n \times n$ respectively.

At any given iteration, SQIC operates in either *variable-reduction mode* or *block-LU mode*. The mode determines which method is used to solve the KKT system. By default, SQIC starts in variable-reduction mode. However, because the variable-reduction method is efficient only when the number of superbasic variables is "small", SQIC will switch to the block-LU method described in Section 5.2 when the number of superbasic variables grows larger than 2000. The user may override the default settings and specify that SQIC start in block-LU mode.

An initial feasible point and basis are found by using the phase 1 algorithm of SQOPT [28], which uses the simplex method to minimize the sum of the infeasibilities of the bound constraints subject to $Ax = b$. The basis defines a vertex with n_S variables temporarily fixed between their bounds. SQIC frees these variables to create a basic set of size $m + n_S$, since the algorithm does not require a vertex to start. If the KKT matrix associated with this basic set has incorrect inertia, then the number of negative eigenvalues is greater than m and the estimated number of temporary constraints e_a is defined as the difference of these numbers. If e_a is greater than $\max\big(10, \frac{1}{2}(n_B - m)\big)$, then the n_S variables are removed from the basic set and the initial m-basis provided by SQOPT is used to define a vertex. Otherwise, the method described in Section 6.2 is used to define temporary constraints that define a second-order-consistent basis.

Three linear solvers have been incorporated into SQIC to store the block-LU (or block-LDL^T) factors of the KKT matrix. These are the symmetric LDL^T solver HSL_MA57 [44], and the unsymmetric LU solvers LUSOL [30] and UMFPACK [11, 12, 13, 14]. The Schur complement matrix is maintained by the dense matrix factorization code LUMOD [52]. LUMOD was updated to Fortran 90 by Huynh [45] for the convex quadratic programming code QPBLU, which also utilizes a block-LU scheme. Modifications were made to the Fortran 90 version of LUMOD to incorporate it into SQIC.

The algorithm for computing temporary constraints for a second-order-consistent basis in Section 6.2 requires a linear solver that computes an LDL^T factorization and provides access to the matrix L. Of the three solvers that were tested, only HSL_MA57 is a symmetric indefinite solver, but it does not provide access to L by default. However, a subroutine returning L was provided by Iain Duff [17], and so HSL_MA57 is the only solver capable of defining temporary constraints. For all other solvers, a vertex is defined if the initial basis is not second-order-consistent.

By default, a scaling of H and A is defined based on the scaling algorithm in [51] applied to the full KKT matrix

$$\begin{pmatrix} H & A^T \\ A & 0 \end{pmatrix}.$$

The built-in scaling routines used by the linear solvers are turned off. When checking for linear dependence in (4.4), SQIC tests

$$[u_B]_r < \epsilon\, [p_B]_r,$$

where ϵ is the tolerance $\epsilon = 5 \times 10^{-9}$.

There are two situations in which the Schur complement is discarded and the KKT matrix is refactorized. The first is for structural reasons when the dimension of the Schur complement exceeds $\min\big(1000, \frac{1}{2}(n_B + m)\big)$. The second is for numerical reasons when the estimated condition number of the Schur complement is greater than 10^{16}. If the refactorization is done for numerical reasons, a step of iterative refinement is used to define updates to x and π. In addition, if the estimated condition number of the Schur complement or the block KKT matrix is greater than 10^9, then the residuals of the KKT systems are checked, and corrections are made to x and π if the norm of the solution is greater than $\epsilon_M^{0.8} \max(\texttt{condB}, \texttt{condU})$, where ϵ_M is machine precision, and condB and condU are the estimated condition numbers of the block matrix and Schur complement, respectively.

For numerical reasons, both SQIC and SQOPT allow the variables (x, s) to stray outside their bounds by as much as a specified tolerance $\delta = 10^{-6}$. The EXPAND procedure of Gill et al. [31] takes advantage of δ to reduce the chance of cycling at a point where the active constraints are nearly linearly dependent. Although there is no guarantee of preventing cycling, the probability is very small (see Hall and McKinnon [43]). The main feature of EXPAND is that over a period of $K = 10^3$ iterations, a "working" feasibility tolerance increases from $\frac{1}{2}\delta$ to δ in steps of $\frac{1}{2}\delta/K$. At certain stages, the following "resetting procedure" is used to remove small constraint infeasibilities. First, all nonbasic variables are moved exactly onto their bounds. A count is kept of the number of non-trivial adjustments made. If the count is nonzero, the basic variables are recomputed. Finally, the working feasibility tolerance is reinitialized to $\frac{1}{2}\delta$. If a problem requires more than K iterations, the resetting procedure is invoked and a new cycle of iterations is started. (The decision to resume phase 1 or phase 2 is based on comparing any infeasibilities with δ.) The resetting procedure is also invoked when the solver reaches an apparently optimal, infeasible, or unbounded solution, unless this situation has already occurred twice. If any non-trivial adjustments are made, iterations are continued.

The EXPAND procedure allows a positive step to be taken at every iteration, and also provides a potential *choice* of constraint to be added to the working set. EXPAND first

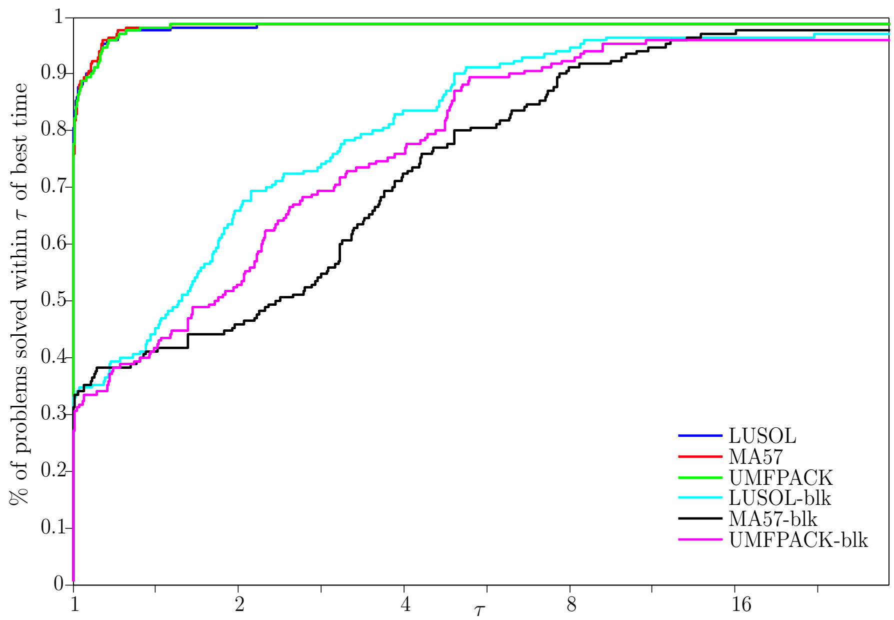

Figure 3: Performance profile of solve times for `SQIC` on the CUTEr QP test set with a small number of superbasics.

computes a maximum feasible step α_P for an expanded feasible defined by perturbing each constraint bound by the working feasibility tolerance. All constraints at a distance α ($\alpha \leq \alpha_P$) along p from the current point are then viewed as acceptable candidates for inclusion in the working set. The constraint whose normal makes the biggest angle with the search direction is added to the working set. This strategy helps keep the the basis matrix A_B well-conditioned.

7.2. Results

A total of 253 quadratic problems were identified from the CUTEr [5, 39] test set. (Linear problems were excluded from the test set since `SQIC` uses the LP algorithm of `SQOPT` to solve them.) The problems are grouped into two sets based on the final number of superbasic variables. A problem is in the "large" set if the final number of superbasics is greater than 1000 or $\frac{1}{2}(m+n)$. The remaining problems form the "small" set. The CUTEr set contains 170 small and 83 large problems. A time limit of 5000 seconds was imposed in each case. (In practice, the 5000 second limit is not exact since the time limit is checked every twenty iterations.)

Results are presented for `SQIC` with its default settings using the three linear solvers `HSL_MA57`, `UMFPACK` and the included solver `LUSOL`, on an iMac with a 2.8GHz Intel Core i7 processor and 16GB of memory. The results are summarized using performance profiles (in $\log_2$ scale) proposed by Dolan and Moré [15]. In addition, all problems were run using only the block-LU method to illustrate the efficiency of the variable-reduction method. These results are denoted by the suffix "-blk".

For all solvers and settings, every problem ran to completion with the exception of `CVXQP1` and `CVXQP3` for `UMFPACK` with the block-LU option. These problems resulted in "ran out of memory" error messages from Mac OS. In the performance profiles, the problems are considered failures, however, these two cases ran to completion on a Linux machine running Ubuntu 12.04 with a 3.33GHz Intel Xeon processor and 16GB of memory.

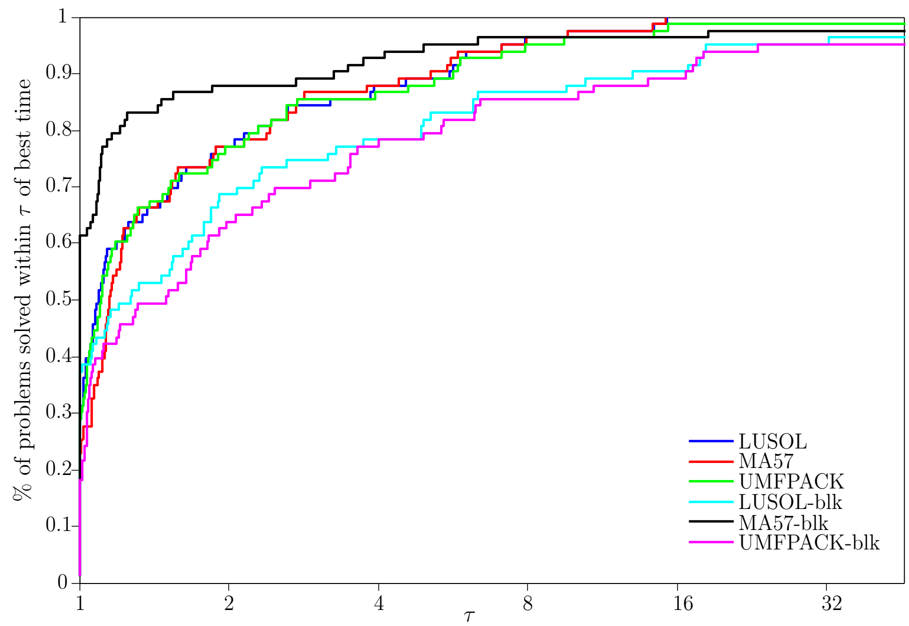

Figure 4: Performance profile of solve times for SQIC on the CUTEr QP test set with a large number of superbasics.

In variable-reduction mode, SQIC hit the iteration limit on problems A5NDSDSDL and A5NDSDSSL with all solvers. With UMFPACK, problem RDW2D52U resulted in a memory error.

In block-matrix mode, problems A5NSDSSL, LEUVEN3 and LEUVEN5 hit iteration limits with every solver. The same issue occurred for problems CVXQP1 and CVXQP3 with HSL_MA57. In addition, problem A5NNDNDL hit the time limit for UMFPACK, while problems A5NNDNSL, A5NSDSDL, RDW2D51U, RDW2D52U and WALL100 hit the time limit for UMFPACK and LUSOL.

Performance profiles for problems with a "small" number of superbasics are shown in Figure 3. It is clear from the profile that starting in variable-reduction mode with any of the three solvers is significantly more effective than using plain block-matrix mode on this set of problems. This is likely due to the overhead of factoring the larger block KKT matrix.

On the problems with a "large" number of superbasics, SQIC in block-matrix mode using HSL_MA57 appears to be the most efficient, solving about 62% of the problems in the best time. HSL_MA57 allows SQIC to start at non-vertices with an arbitrary number of superbasic variables, giving it an advantage over the other solvers, which must start at a vertex with no superbasic variables.

Results are also presented that compare SQIC with the convex QP solver SQOPT [28], which is an implementation of a reduced-Hessian, reduced-gradient active-set method. The method of SQOPT removes a variable from the nonbasic set at the *start* of a sequence of intermediate iterates and maintains the matrix factors associated with the variable reduction method described in Section 5.1. With this method, the reduced Hessian Z^THZ is positive semidefinite with at most one zero eigenvalue. If the reduced Hessian is positive definite, a suitable direction is computed from the equations

$$Z^THZp_S = -Z^Tg, \tag{7.1}$$

which are solved using a dense Cholesky factor of Z^THZ. If the reduced Hessian is singular, the Cholesky factor is used to define p_S such that $Z^THZp_S = 0$ and $p_S^TZ^Tg < 0$. If the number of superbasics is large, then solving (7.1) becomes expensive. By default, SQOPT switches to a conjugate-gradient method to solve for a direction, when n_S is greater than 2000. Therefore, it is to be expected that SQIC, which utilizes the block-LU method, will provide superior performance when there are many superbasics.

Figures 5 and 6 are the performance profiles of SQIC and SQOPT on convex CUTEr problems with a small and large number of superbasics. 141 convex problems were identified, with 66 in the "small" set and 75 in the "large" set. The choice of linear solver does not have a big effect on the performance of SQIC on convex problems. However, it is clear from Figure 5 that SQOPT is the best solver for convex problems with a "small" number of superbasics, while SQIC is the better choice for problems with a large number of superbasics.

8. Summary

Numerical results suggest that SQIC is a more effective solver on problems with a large number of degrees of freedom than the convex solver SQOPT. In addition, SQIC is able to solve nonconvex problems, and can incorporate third-party linear solvers.

Future research will consider the application and implementation of the method to the dual of a convex QP, as well as applying the algorithm as an augmented Lagrangian method.

Acknowledgments

We extend sincere thanks to Michael Saunders, Iain Duff and Nick Gould for their assistance during the development of SQIC and the computational tests of Section 7.

References

[1] P. R. Amestoy, I. S. Duff, J.-Y. L'Excellent, and J. Koster. A fully asynchronous multifrontal solver using distributed dynamic scheduling. *SIAM J. Matrix Anal. Appl.*, 23(1):15–41 (electronic), 2001. 21

[2] C. Ashcraft and R. Grimes. SPOOLES: an object-oriented sparse matrix library. In *Proceedings of the Ninth SIAM Conference on Parallel Processing for Scientific Computing 1999 (San Antonio, TX)*, page 10, Philadelphia, PA, 1999. SIAM. 21

[3] R. A. Bartlett and L. T. Biegler. QPSchur: a dual, active-set, Schur-complement method for large-scale and structured convex quadratic programming. *Optim. Eng.*, 7(1):5–32, 2006. 3

[4] J. Bisschop and A. Meeraus. Matrix augmentation and partitioning in the updating of the basis inverse. *Math. Program.*, 13:241–254, 1977. 21

[5] I. Bongartz, A. R. Conn, N. I. M. Gould, and P. L. Toint. CUTE: Constrained and unconstrained testing environment. *ACM Trans. Math. Software*, 21(1):123–160, 1995. 3, 27

[6] J. M. Borwein. Necessary and sufficient conditions for quadratic minimality. *Numer. Funct. Anal. and Optimiz.*, 5:127–140, 1982. 4

[7] G. G. Brown and G. W. Graves. Elastic programming: A new approach to large-scale mixed-integer optimization, November 1975. Presented at the presented at ORSA/TIMS meeting, Las Vegas, NV.

[8] J. R. Bunch and L. Kaufman. A computational method for the indefinite quadratic programming problem. *Linear Algebra Appl.*, 34:341–370, 1980. 3

[9] L. B. Contesse. Une caractérisation complète des minima locaux en programmation quadratique. *Numer. Math.*, 34:315–332, 1980. 4

[10] R. W. Cottle, G. J. Habetler, and C. E. Lemke. On classes of copositive matrices. *Linear Algebra Appl.*, 3:295–310, 1970. 4

[11] T. A. Davis. Algorithm 832: UMFPACK V4.3—an unsymmetric-pattern multifrontal method. *ACM Trans. Math. Software*, 30(2):196–199, 2004. 26

[12] T. A. Davis. A column pre-ordering strategy for the unsymmetric-pattern multifrontal method. *ACM Trans. Math. Software*, 30(2):167–195, 2004. 26

[13] T. A. Davis and I. S. Duff. An unsymmetric-pattern multifrontal method for sparse *LU* factorization. *SIAM J. Matrix Anal. Appl.*, 18(1):140–158, 1997. 26

[14] T. A. Davis and I. S. Duff. A combined unifrontal/multifrontal method for unsymmetric sparse matrices. *ACM Trans. Math. Software*, 25(1):1–20, 1999. 26

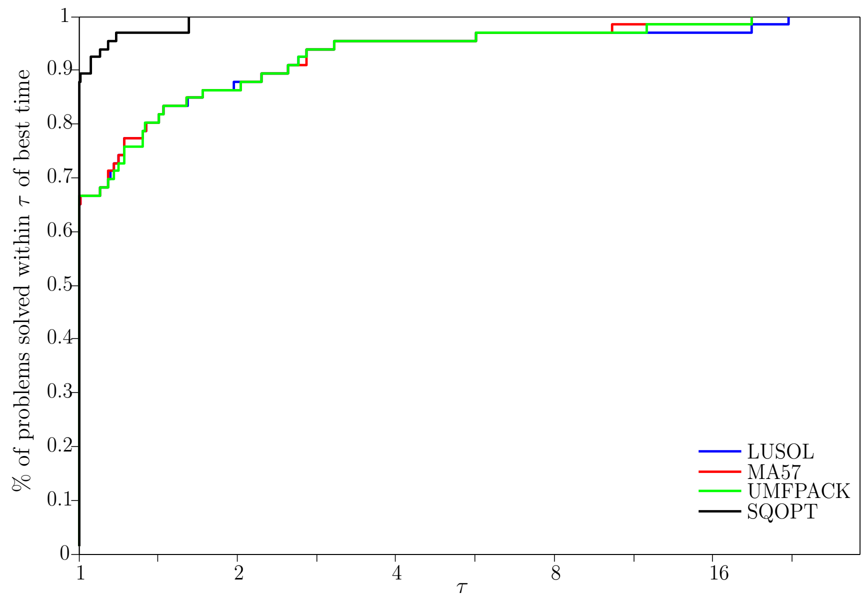

Figure 5: Performance profile of solve times for SQIC and SQOPT on convex CUTEr problems with a small number of superbasics.

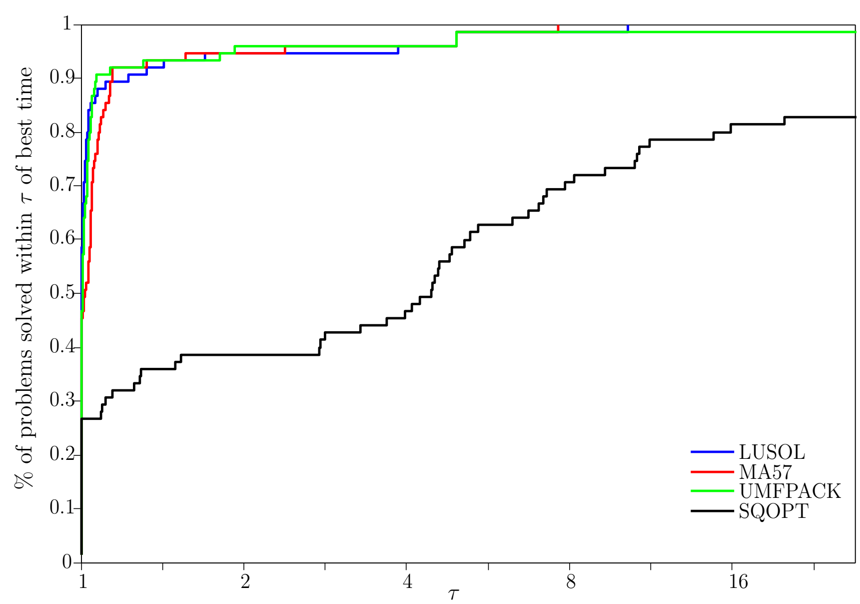

Figure 6: Performance profile of solve times for SQIC and SQOPT on convex CUTEr problems with a large number of superbasics.

[15] E. D. Dolan and J. J. Moré. Benchmarking optimization software with COPS. Technical Memorandum ANL/MCS-TM-246, Argonne National Laboratory, Argonne, IL, 2000. 27

[16] I. S. Duff. MA57—a code for the solution of sparse symmetric definite and indefinite systems. *ACM Trans. Math. Software*, 30(2):118–144, 2004. 21

[17] I. S. Duff. Subroutine `ma57lfd.f`, October 2010. Private communication. 26

[18] I. S. Duff and J. K. Reid. The multifrontal solution of indefinite sparse symmetric linear equations. *ACM Trans. Math. Software*, 9:302–325, 1983. 21

[19] S. K. Eldersveld and M. A. Saunders. A block-LU update for large-scale linear programming. *SIAM J. Matrix Anal. Appl.*, 13:191–201, 1992. 22

[20] R. Fletcher. A general quadratic programming algorithm. *J. Inst. Math. Applics.*, 7:76–91, 1971. 2, 6

[21] A. Forsgren, P. E. Gill, and W. Murray. On the identification of local minimizers in inertia-controlling methods for quadratic programming. *SIAM J. Matrix Anal. Appl.*, 12:730–746, 1991. 5

[22] M. P. Friedlander and S. Leyffer. Global and finite termination of a two-phase augmented Lagrangian filter method for general quadratic programs. *SIAM J. Sci. Comput.*, 30(4):1706–1729, 2008. 3

[23] P. E. Gill. Recent developments in numerically stable methods for linear programming. *IMA Bulletin*, 10:180–186, 1973. 15

[24] P. E. Gill, N. I. M. Gould, W. Murray, M. A. Saunders, and M. H. Wright. A weighted Gram-Schmidt method for convex quadratic programming. *Math. Program.*, 30:176–195, 1984. 3

[25] P. E. Gill and W. Murray. Numerically stable methods for quadratic programming. *Math. Program.*, 14:349–372, 1978. 2

[26] P. E. Gill, W. Murray, and M. A. Saunders. User's guide for SQOPT 5.3: a Fortran package for large-scale linear and quadratic programming. Numerical Analysis Report 97-4, Department of Mathematics, University of California, San Diego, La Jolla, CA, 1997. 3

[27] P. E. Gill, W. Murray, and M. A. Saunders. SNOPT: An SQP algorithm for large-scale constrained optimization. *SIAM Rev.*, 47:99–131, 2005. 23

[28] P. E. Gill, W. Murray, and M. A. Saunders. User's guide for SQOPT Version 7: Software for large-scale linear and quadratic programming. Numerical Analysis Report 06-1, Department of Mathematics, University of California, San Diego, La Jolla, CA, 2006. 25, 28

[29] P. E. Gill, W. Murray, M. A. Saunders, and M. H. Wright. Sparse matrix methods in optimization. *SIAM J. Sci. Statist. Comput.*, 5(3):562–589, 1984. 22

[30] P. E. Gill, W. Murray, M. A. Saunders, and M. H. Wright. Maintaining *LU* factors of a general sparse matrix. *Linear Algebra Appl.*, 88/89:239–270, 1987. 26

[31] P. E. Gill, W. Murray, M. A. Saunders, and M. H. Wright. A practical anti-cycling procedure for linearly constrained optimization. *Math. Program.*, 45:437–474, 1989. 26

[32] P. E. Gill, W. Murray, M. A. Saunders, and M. H. Wright. A Schur-complement method for sparse quadratic programming. In M. G. Cox and S. J. Hammarling, editors, *Reliable Numerical Computation*, pages 113–138. Oxford University Press, 1990. 2, 21

[33] P. E. Gill, W. Murray, M. A. Saunders, and M. H. Wright. Inertia-controlling methods for general quadratic programming. *SIAM Rev.*, 33(1):1–36, 1991. 2

[34] P. E. Gill, W. Murray, and M. H. Wright. *Numerical Linear Algebra and Optimization, volume 1*. Addison-Wesley Publishing Company, Redwood City, 1991. 5

[35] P. E. Gill and E. Wong. Sequential quadratic programming methods. In J. Lee and S. Leyffer, editors, *Mixed Integer Nonlinear Programming*, volume 154 of *The IMA Volumes in Mathematics and its Applications*, pages 147–224. Springer New York, 2012. 10.1007/978-1-4614-1927-3_6. 2

[36] D. Goldfarb and A. Idnani. A numerically stable dual method for solving strictly convex quadratic programs. *Math. Programming*, 27(1):1–33, 1983. 3

[37] N. I. M. Gould. On practical conditions for the existence and uniqueness of solutions to the general equality quadratic programming problem. *Math. Program.*, 32:90–99, 1985. 6

[38] N. I. M. Gould. An algorithm for large-scale quadratic programming. *IMA J. Numer. Anal.*, 11(3):299–324, 1991. 2

[39] N. I. M. Gould, D. Orban, and P. L. Toint. CUTEr and SifDec: A constrained and unconstrained testing environment, revisited. *ACM Trans. Math. Software*, 29(4):373–394, 2003. 3, 27

[40] N. I. M. Gould, D. Orban, and P. L. Toint. ***GALAHAD***, a library of thread-safe Fortran 90 packages for large-scale nonlinear optimization. *ACM Trans. Math. Software*, 29(4):353–372, Dec 2003. 3

[41] N. I. M. Gould and P. L. Toint. An iterative working-set method for large-scale nonconvex quadratic programming. *Appl. Numer. Math.*, 43(1-2):109–128, 2002. 19th Dundee Biennial Conference on Numerical Analysis (2001). 3

[42] N. I. M. Gould and P. L. Toint. Numerical methods for large-scale non-convex quadratic programming. In *Trends in industrial and applied mathematics (Amritsar, 2001)*, volume 72 of *Appl. Optim.*, pages 149–179. Kluwer Acad. Publ., Dordrecht, 2002. 3

[43] J. A. J. Hall and K. I. M. McKinnon. The simplest examples where the simplex method cycles and conditions where EXPAND fails to prevent cycling. Technical Report MS 96-010, Department of Mathematics and Statistics, University of Edinburgh, 1996. 26

[44] HSL. A collection of Fortran codes for large-scale scientific computation. http://www.hsl.rl.ac.uk, 2011. 26

[45] H. M. Huynh. *A Large-Scale Quadratic Programming Solver Based on Block-LU Updates of the KKT System.* PhD thesis, Program in Scientific Computing and Computational Mathematics, Stanford University, Stanford, CA, 2008. 22, 26

[46] A. Majthay. Optimality conditions for quadratic programming. *Math. Programming*, 1:359–365, 1971. 4

[47] J. Nocedal and S. J. Wright. *Numerical Optimization.* Springer-Verlag, New York, 1999. 5

[48] P. M. Pardalos and G. Schnitger. Checking local optimality in constrained quadratic programming is NP-hard. *Oper. Res. Lett.*, 7(1):33–35, 1988. 4

[49] P. M. Pardalos and S. A. Vavasis. Quadratic programming with one negative eigenvalue is NP-hard. *J. Global Optim.*, 1(1):15–22, 1991. 4

[50] M. J. D. Powell. On the quadratic programming algorithm of Goldfarb and Idnani. *Math. Programming Stud.*, (25):46–61, 1985. 3

[51] D. Ruiz. A scaling algorithm to equilibrate both row and column norms in matrices. Report RAL-TR-2001-034, Rutherford Appleton Laboratory, Oxon, UK, September 2001. 26

[52] M. A. Saunders. LUMOD: Updating a dense square factorization LC = U. http://www.stanford.edu/group/SOL/software/lumod.html. 26

[53] O. Schenk and K. Gärtner. Solving unsymmetric sparse systems of linear equations with PARDISO. In *Computational science—ICCS 2002, Part II (Amsterdam)*, volume 2330 of *Lecture Notes in Comput. Sci.*, pages 355–363. Springer, Berlin, 2002. 21

[54] J. A. Tomlin. On pricing and backward transformation in linear programming. *Math. Programming*, 6:42–47, 1974. 15

[55] E. Wong. *Active-Set Methods for Quadratic Programming.* PhD thesis, Department of Mathematics, University of California, San Diego, June 2011. 14

9. Appendix

Tables 1–12 provide more detailed information about the runs used to compile the performance profiles of Figures 3–6. For each problem, the tables list the following: the number of linear constraints "m"; the number of variables "n"; the final objective value "Objective"; the final residual norm of the equality constraints "Inf"; the number of iterations taken "Itn"; the final number of superbasics "nS"; the number of factors of the block matrix (in block-matrix mode) "bkFac"; and the number of factors of the basis matrix (in variable-reduction mode) "nFac". The last column lists the total number of seconds for the problem to run "Time".

Superscripts on a problem name indicate a non-optimal exit. Superscripts i and u mark the problems judged to be infeasible or unbounded, respectively; t, r and n mark problems that hit the time limit, restart iteration limit, or total iteration limit, respectively; and m indicates a memory error. The superscript $**$ denotes problems that failed to run to completion on the iMac. The results listed for these problems are from a Linux machine running Ubuntu 12.04 with a 3.33GHz Intel Xeon processor and 16GB of memory.

Table 1: Results for SQIC with LUSOL on ''small'' CUTEr QPs

Name	m	n	Objective	$\|Ax-b\|$	Itn	nS	bkFac	nFac	Time
A0ENDNDL	15002	45006	0.0000E+00	1.2049E-11	7276	0	0	152	25.34
A0ENINDL	15002	45006	0.0000E+00	3.4195E-12	7226	0	0	149	24.57
A0ENSNDL	15002	45006	-2.2319E-09	3.1831E-09	5781	0	0	120	20.49
A0ESDNDL	15002	45006	0.0000E+00	5.2419E-12	7192	0	0	140	26.01
A0ESINDL	15002	45006	0.0000E+00	2.1649E-12	7165	0	0	147	24.09
A0ESSNDL	15002	45006	-2.2519E-09	5.4170E-09	5865	0	0	121	22.55
A0NNDNDL	20004	60012	0.0000E+00	1.1623E-12	61929	0	0	1261	404.69
A0NNDNIL	20004	60012	4.8858E+01	4.4889E-12	12155	61	0	244	53.85
A0NNDNSL	20004	60012	-6.6875E-09	9.2523E-12	39329	0	0	795	209.22
A0NNSNSL	20004	60012	-1.1023E-08	6.3086E-12	21119	0	0	431	97.21
A0NSDSDL	20004	60012	0.0000E+00	8.8097E-12	30797	0	0	629	149.06
A0NSDSDS	2004	6012	-3.8600E-09	2.0915E-07	1615	0	0	37	0.56
A0NSDSIL	20004	60012	0.0000E+00	3.3788E-11	13753	0	0	281	59.60
A0NSDSSL	20004	60012	-9.3001E-11	1.2024E-11	23318	0	0	477	93.61
A0NSSSSL	20004	60012	-1.5102E-08	6.3385E-12	17173	0	0	352	66.76
A2ENDNDL	15002	45006	0.0000E+00	4.6408E-12	7117	0	0	143	26.12
A2ENINDL	15002	45006	5.6153E-26	3.6365E-12	7114	2	0	143	26.55
A2ENSNDL	15002	45006	-2.9789E-09	6.1114E-12	5568	1	0	115	21.09
A2ESDNDL	15002	45006	9.6115E-27	5.2652E-13	6975	8	0	132	26.43
A2ESINDL	15002	45006	1.4546E-27	3.7362E-13	7009	4	0	142	26.24
A2ESSNDL	15002	45006	-2.9789E-09	6.7244E-12	5422	2	0	112	20.84
A2NNDNDL	20004	60012	4.4271E-11	5.8288E-08	71795	21	0	1450	513.67
A2NNDNIL^{i}	20004	60012	0.0000E+00	6.4845E-14	11717	0	0	241	49.44
A2NNDNSL	20004	60012	0.0000E+00	2.7387E-12	44025	0	0	893	266.74
A2NNSNSL	20004	60012	-2.5420E-09	7.4080E-08	22765	0	0	465	86.68
A2NSDSDL	20004	60012	-2.7289E-12	6.3848E-08	50809	8	0	1055	324.34
A2NSDSIL	20004	60012	5.3679E+01	7.4628E-08	15050	253	0	301	86.97
A2NSDSSL	20004	60012	-3.2536E-12	7.4640E-12	29605	0	0	654	167.93
A2NSSSSL	20004	60012	-8.7014E-05	2.1586E-07	20498	0	0	419	77.83
A5ENDNDL	15002	45006	4.8867E-26	3.7520E-09	6606	18	0	128	25.27
A5ENINDL	15002	45006	-2.7579E-29	7.7338E-09	6613	3	0	130	25.35
A5ENSNDL	15002	45006	-1.1200E-09	8.3769E-12	5111	0	0	105	19.86
A5ESDNDL	15002	45006	2.7862E-26	1.0486E-13	6442	6	0	128	27.44
A5ESINDL	15002	45006	5.9148E-27	6.4776E-12	6430	3	0	130	26.20
A5ESSNDL	15002	45006	-1.1317E-09	7.6808E-12	4785	0	0	97	14.77
A5NNDNDL	20004	60012	3.8603E+01	1.6980E-12	64923	183	0	1348	465.04
A5NNDNIL^{i}	20004	60012	0.0000E+00	9.3408E-14	10259	0	0	211	42.33
A5NNDNSL	20004	60012	1.7860E-11	2.0308E-11	41977	0	0	867	245.50
A5NNSNSL	20004	60012	-2.2752E-09	2.6288E-07	25628	0	0	519	125.89
A5NSDSDL^{r}	20004	60012	1.0300E+04	7.1406E-13	47007	1	0	1025	384.05
A5NSDSDM	2004	6012	-3.8600E-09	2.0915E-07	1615	0	0	37	0.63

Table 1: Results for SQIC with LUSOL on ''small'' CUTEr QPs (continued)

Name	m	n	Objective	$\|Ax - b\|$	Itn	nS	bkFac	nFac	Time
A5NSDSIL	20004	60012	1.1413E+01	2.5155E-07	11945	614	0	230	75.91
A5NSDSSL^r	20004	60012	4.4189E+04	1.2859E-11	29762	1	0	666	208.95
A5NSSNSM	2004	6012	-3.8600E-09	2.0915E-07	1615	0	0	37	0.64
A5NSSSSL	20004	60012	-7.8358E-09	3.2229E-07	22517	0	0	460	109.25
AVGASA	10	8	-4.6319E+00	8.4581E-12	11	3	0	1	0.00
AVGASB	10	8	-4.4832E+00	1.3672E-16	11	3	0	1	0.00
BIGGSC4	7	4	-2.4375E+01	9.8000E-12	12	1	0	1	0.00
BLOCKQP1	5001	10010	-4.9940E+03	1.8622E-16	5014	9	0	2	4.96
BLOCKQP2	5001	10010	-4.9928E+03	3.0600E-08	7516	9	0	104	13.44
BLOCKQP3	5001	10010	-2.4950E+03	1.8223E-16	5014	9	0	2	5.31
BLOCKQP4	5001	10010	-2.4933E+03	3.0600E-08	8493	9	0	104	18.74
BLOCKQP5	5001	10010	-2.4950E+03	1.7768E-16	5020	9	0	2	4.14
BQP1VAR	0	1	0.0000E+00	0.0000E+00	2	0	0	1	0.00
CVXBQP1	0	10000	2.2502E+06	1.8646E-17	10001	0	0	1	7.99
CVXQP3	7500	10000	1.1571E+08	1.7545E-09	11418	390	0	196	11.60
DEGENQP	8010	20	0.0000E+00	0.0000E+00	12	0	0	2	0.02
DUALC1	215	9	6.1553E+03	3.5126E-15	5	2	0	2	0.00
DUALC2	229	7	3.5513E+03	2.1456E-15	4	2	0	2	0.00
DUALC5	278	8	4.2723E+02	9.8919E-16	5	4	0	2	0.00
DUALC8	503	8	1.8309E+04	2.1421E-15	7	2	0	2	0.00
FERRISDC	210	2200	0.0000E+00	0.0000E+00	1	0	0	2	0.26
GENHS28	8	10	9.2717E-01	1.8625E-15	3	2	0	2	0.00
GMNCASE1	300	175	2.6697E-01	3.8875E-16	102	53	0	1	0.03
GMNCASE2	1050	175	-9.9444E-01	3.6566E-15	104	46	0	2	0.04
GMNCASE3	1050	175	1.5251E+00	1.2018E-14	107	48	0	2	0.04
GMNCASE4	350	175	5.9469E+03	7.7433E-12	141	0	0	3	0.06
GOULDQP1	17	32	-3.4853E+03	1.2250E-11	23	0	0	2	0.00
GOULDQP2	9999	19999	1.8512E-12	1.4803E-16	1	0	0	2	0.88
HARKERP2	0	1000	-5.0000E-01	0.0000E+00	1000	0	0	1	4.47
HATFLDH	7	4	-2.4500E+01	0.0000E+00	4	0	0	1	0.00
HS118	17	15	6.6482E+02	1.2250E-11	23	0	0	1	0.00
HS21	1	2	-9.9960E+01	1.7764E-15	2	1	0	1	0.00
HS268	5	5	-7.2760E-12	2.5794E-15	11	5	0	1	0.00
HS3	0	2	0.0000E+00	0.0000E+00	3	1	0	1	0.00
HS35	1	3	1.1111E-01	3.4648E-11	6	2	0	1	0.00
HS35I	1	3	1.1111E-01	3.4648E-11	6	2	0	1	0.00
HS35MOD	1	3	2.5000E-01	0.0000E+00	2	1	0	1	0.00
HS3MOD	0	2	2.2187E-31	0.0000E+00	3	1	0	1	0.00
HS44	6	4	-1.3000E+01	2.6585E-16	3	0	0	1	0.00
HS44NEW	6	4	-1.3000E+01	3.9878E-16	6	0	0	1	0.00
HS51	3	5	-8.8818E-16	1.5632E-16	3	2	0	2	0.00
HS52	3	5	5.3266E+00	1.0747E-16	3	2	0	2	0.00
HS53	3	5	4.0930E+00	1.5632E-16	3	2	0	2	0.00
HS76	3	4	-4.6818E+00	3.0304E-16	5	2	0	1	0.00
HS76I	3	4	-4.6818E+00	3.0304E-16	5	2	0	1	0.00
KSIP	1001	20	5.7580E-01	1.4470E-17	1536	18	0	65	0.86
LEUVEN1	2220	1530	-1.5243E+07	7.2852E-09	1516	12	0	31	0.42
LEUVEN2	2329	1530	-1.4147E+07	1.6956E-08	610	2	0	12	0.16
LEUVEN3	2973	1200	-1.0381E+09	6.0444E-09	1079	50	0	29	3.15
LEUVEN4	2973	1200	-1.4083E+09	2.1971E-11	1709	50	0	71	6.80
LEUVEN5	2973	1200	-1.0381E+09	6.0444E-09	1079	50	0	29	3.15
LEUVEN6	3091	1200	-1.4533E+08	1.6050E-09	578	30	0	26	2.57
LEUVEN7	946	360	6.9455E+02	2.4889E-12	203	19	0	4	0.12
LINCONT^i	419	1257	0.0000E+00	4.6663E-14	126	0	0	4	0.03
LISWET1	10000	10002	3.6121E+01	1.3323E-15	4	2	0	2	0.15
LISWET10	10000	10002	4.9483E+01	1.1435E-14	49	16	0	4	0.24
LISWET11	10000	10002	4.9524E+01	4.9220E-15	43	30	0	4	0.24
LISWET12	10000	10002	1.7369E+03	9.2519E-16	28	5	0	4	0.21
LISWET2	10000	10002	2.5000E+01	1.2212E-15	22	4	0	2	0.18
LISWET3	10000	10002	2.5000E+01	2.8630E-08	444	261	0	5	0.86

Table 1: Results for SQIC with LUSOL on ‘‘small’’ CUTEr QPs (continued)

Name	m	n	Objective	$\|Ax - b\|$	Itn	nS	bkFac	nFac	Time
LISWET4	10000	10002	2.5000E+01	2.0362E-07	431	270	0	5	0.85
LISWET5	10000	10002	2.5000E+01	4.9305E-08	412	254	0	5	0.78
LISWET6	10000	10002	2.5000E+01	2.3685E-15	341	222	0	5	0.67
LISWET7	10000	10002	4.9884E+02	7.7716E-16	4	2	0	2	0.15
LISWET8	10000	10002	7.1447E+02	6.6613E-16	22	13	0	2	0.19
LISWET9	10000	10002	1.9632E+03	1.7023E-15	16	4	0	4	0.20
LOTSCHD	7	12	2.3984E+03	3.4710E-15	8	0	0	2	0.00
MARATOSB^u	0	2	-1.4400E+06	2.0016E-17	2	0	0	1	0.00
MPC1	3833	2550	-2.3262E+07	1.8844E-08	1362	0	0	26	0.52
MPC10	2351	1530	-1.5034E+07	4.8004E-08	1180	11	0	24	0.35
MPC11	2351	1530	-1.5030E+07	9.1770E-09	924	34	0	17	0.27
MPC12	2351	1530	-1.5033E+07	1.9375E-07	1155	19	0	22	0.35
MPC13	2351	1530	-1.5034E+07	4.9520E-09	1070	13	0	21	0.31
MPC14	2351	1530	-1.5034E+07	3.1434E-07	1190	16	0	22	0.34
MPC15	2351	1530	-1.5034E+07	2.1451E-06	997	15	0	19	0.29
MPC16	2351	1530	-1.5034E+07	8.5862E-07	1047	16	0	20	0.30
MPC2	2351	1530	-1.5033E+07	9.2528E-08	1201	27	0	21	0.35
MPC3	2351	1530	-1.5030E+07	2.3069E-08	1228	32	0	23	0.39
MPC4	2351	1530	-1.5033E+07	4.7017E-09	1259	21	0	24	0.36
MPC5	2351	1530	-1.5033E+07	9.2840E-08	1271	25	0	23	0.34
MPC6	2351	1530	-1.5034E+07	4.7094E-09	1179	18	0	22	0.35
MPC7	2351	1530	-1.5034E+07	2.3473E-09	1087	15	0	21	0.31
MPC8	2351	1530	-1.5034E+07	5.1577E-09	1158	13	0	22	0.35
MPC9	2351	1530	-1.5034E+07	1.9332E-07	1185	11	0	23	0.35
NASH^i	24	72	0.0000E+00	9.4369E-16	2	0	0	2	0.00
NCVXBQP1	0	10000	-1.9855E+10	0.0000E+00	10015	0	0	1	8.24
NCVXBQP2	0	10000	-1.3245E+10	1.8604E-17	11223	51	0	1	9.24
NCVXBQP3	0	10000	-6.4122E+09	2.3813E-15	10837	126	0	1	8.74
NCVXQP1	500	1000	-7.1562E+07	6.8979E-11	749	0	0	8	0.09
NCVXQP2	500	1000	-5.7759E+07	4.2449E-11	1057	0	0	16	0.15
NCVXQP3	500	1000	-2.9253E+07	3.6206E-11	1223	19	0	13	0.20
NCVXQP4	250	1000	-9.3995E+07	1.0037E-11	788	0	0	4	0.09
NCVXQP5	250	1000	-6.6257E+07	1.0746E-11	824	0	0	5	0.09
NCVXQP6	250	1000	-3.4172E+07	6.9472E-12	926	49	0	6	0.12
NCVXQP7	750	1000	-4.3521E+07	7.4509E-11	660	0	0	11	0.08
NCVXQP8	750	1000	-3.0103E+07	9.6553E-11	909	0	0	16	0.12
NCVXQP9	750	1000	-2.1230E+07	2.6031E-09	961	11	0	15	0.16
PENTDI	0	5000	-7.5000E-01	0.0000E+00	3	2	0	1	0.07
PORTSNQP	2	100000	3.3332E+04	3.5313E-14	108265	257	0	882	42.93
PORTSQP	1	100000	3.3331E+04	1.8822E-11	100317	315	0	2	23.96
POWELL20	10000	10000	5.2090E+10	1.6977E-11	5003	1	0	103	6.08
PRIMAL1	85	325	-3.5013E-02	1.5669E-12	218	133	0	2	0.02
PRIMAL2	96	649	-3.3734E-02	3.0138E-16	408	302	0	2	0.09
PRIMALC1	9	230	-6.1553E+03	1.4648E-12	19	14	0	1	0.00
PRIMALC2	7	231	-3.5513E+03	1.1415E-13	4	1	0	1	0.00
PRIMALC5	8	287	-4.2723E+02	3.1580E-14	10	5	0	1	0.00
PRIMALC8	8	520	-1.8309E+04	1.5526E-10	22	17	0	1	0.00
QPBAND	5000	10000	-9.9992E+03	1.8821E-11	29960	39	0	204	23.50
QPCBLEND	74	83	-7.8425E-03	1.5447E-09	76	2	0	4	0.00
QPCBOEI1	351	384	1.1504E+07	1.2362E-10	701	113	0	9	0.05
QPCBOEI2	166	143	8.1720E+06	5.0579E-11	203	32	0	4	0.01
QPCSTAIR	356	467	6.2044E+06	2.8087E-12	304	20	0	8	0.03
QPNBAND	5000	10000	-4.9997E+04	7.2371E-16	15000	1	0	103	12.18
QPNBLEND	74	83	-8.7056E-03	1.8434E-11	72	3	0	2	0.00
QPNBOEI1	351	384	6.7367E+06	1.9434E-10	685	92	0	11	0.05
QPNBOEI2	166	143	1.3683E+06	5.7753E-11	229	27	0	5	0.01
QPNSTAIR	356	467	5.1460E+06	1.1941E-11	349	20	0	6	0.03
QUDLIN	0	5000	-1.2500E+09	0.0000E+00	5000	0	0	1	0.84
RDW2D51F	65025	132098	1.1353E-03	2.9003E-07	2258	0	0	33	1491.59
RDW2D52F	49	162	8.6159E-03	1.3853E-15	71	37	0	3	0.00

Table 1: Results for SQIC with LUSOL on ''small'' CUTEr QPs (continued)

Name	m	n	Objective	$\|Ax-b\|$	Itn	nS	bkFac	nFac	Time
S268	5	5	-7.2760E-12	2.5794E-15	11	5	0	1	0.00
SIM2BQP	0	2	0.0000E+00	0.0000E+00	2	0	0	1	0.00
SIMBQP	0	2	3.4667E-31	0.0000E+00	3	1	0	1	0.00
SOSQP1	10001	20000	-2.4500E-11	1.5765E-11	3	0	0	2	0.67
STATIC3^u	96	434	-3.0892E+02	0.0000E+00	3	1	0	2	0.00
STEENBRA	108	432	1.6958E+04	8.1667E-11	87	11	0	3	0.01
TAME	1	2	0.0000E+00	6.0309E-17	2	1	0	2	0.00
YAO	2000	2002	1.9770E+02	9.2519E-17	3	1	0	1	0.01
ZECEVIC2	2	2	-4.1250E+00	0.0000E+00	3	1	0	1	0.00

Table 2: Results for SQIC with LUSOL on ''large'' CUTEr QPs

Name	m	n	Objective	$\|Ax-b\|$	Itn	nS	bkFac	nFac	Time
ALLINQP	25000	50000	-5.4813E+03	3.4979E-15	18292	9820	2	75	128.50
AUG2D	10000	20200	1.6874E+06	4.5078E-12	10195	10192	9	3	70.44
AUG2DC	10000	20200	1.8184E+06	9.1939E-10	2003	10200	1	3	13.06
AUG2DCQP	10000	20200	6.4981E+06	3.4838E-10	14461	9994	8	83	84.66
AUG2DQP	10000	20200	6.2370E+06	1.6333E-11	14438	9801	8	78	85.33
AUG3D	8000	27543	2.4561E+04	1.3847E-10	16912	16909	16	3	161.92
AUG3DC	8000	27543	2.7654E+04	1.9647E-09	2003	19543	1	3	19.07
AUG3DCQP	8000	27543	6.1560E+04	3.6515E-08	22202	17665	16	82	196.24
AUG3DQP	8000	27543	5.4229E+04	1.6890E-11	18543	13712	12	83	144.36
BIGGSB1	0	5000	1.5000E-02	0.0000E+00	5005	4998	3	2	12.67
BLOWEYA	2002	4002	-2.2781E-02	7.3505E-14	2002	2000	0	2	6.00
BLOWEYB	2002	4002	-1.5226E-02	9.3323E-14	2002	2000	0	2	6.01
BLOWEYC	2002	4002	-1.5246E-02	1.5728E-12	2003	2000	0	2	6.01
BQPGABIM	0	50	-3.7903E-05	0.0000E+00	41	36	0	1	0.00
BQPGASIM	0	50	-5.5198E-05	0.0000E+00	45	40	0	1	0.00
BQPGAUSS	0	2003	-3.6258E-01	8.4349E-17	2236	1909	0	1	6.33
CHENHARK	0	5000	-2.0000E+00	1.0821E-17	6003	3000	1	2	7.23
CVXQP1	5000	10000	1.0870E+08	6.9982E-11	11491	1261	0	121	16.85
CVXQP2	2500	10000	8.1842E+07	2.5263E-13	8298	2210	1	25	24.73
DIXON3DQ	0	10000	4.4409E-16	0.0000E+00	2003	10000	1	2	7.94
DQDRTIC	0	5000	0.0000E+00	0.0000E+00	2003	5000	1	2	5.59
DTOC3	9998	14999	2.3526E+02	1.0338E-12	2004	4999	1	3	13.05
DUAL1	1	85	3.5013E-02	1.3715E-16	77	62	0	2	0.01
DUAL2	1	96	3.3734E-02	1.5379E-16	94	91	0	2	0.01
DUAL3	1	111	1.3576E-01	2.8691E-16	111	96	0	2	0.01
DUAL4	1	75	7.4609E-01	2.2421E-16	64	61	0	2	0.00
GOULDQP3	9999	19999	2.3796E-05	1.4522E-13	5899	4988	3	19	32.31
GRIDNETA	6724	13284	3.0498E+02	4.3835E-14	2083	2218	1	4	8.35
GRIDNETB	6724	13284	1.4332E+02	1.1769E-11	2003	6561	1	3	9.84
GRIDNETC	6724	13284	1.4832E+02	3.4888E-13	3930	4533	2	4	18.94
HILBERTA	0	10	9.6007E-09	0.0000E+00	8	7	0	1	0.00
HILBERTB	0	50	5.2590E-28	9.5890E-16	51	50	0	1	0.00
HUES-MOD	2	10000	3.4824E+07	2.6831E-12	3828	9444	1	3	8.49
HUESTIS	2	10000	3.4824E+11	1.7966E-11	3827	9444	1	3	8.48
JNLBRNG1	0	15625	-1.8058E-01	0.0000E+00	4020	10247	2	2	18.36
JNLBRNG2	0	15625	-4.1496E+00	0.0000E+00	3639	9139	2	2	15.51
JNLBRNGA	0	15625	-2.6851E-01	0.0000E+00	9971	9968	8	2	40.47
JNLBRNGB	0	15625	-6.2807E+00	0.0000E+00	8480	8477	7	2	33.32
MOSARQP1	700	2500	-3.8214E+03	6.6779E-11	3254	1021	0	23	3.75
MOSARQP2	700	2500	-5.0526E+03	4.4410E-14	2553	1640	0	6	4.01
NOBNDTOR	0	14884	-4.4054E-01	0.0000E+00	5899	12078	4	2	27.54
OBSTCLAE	0	15625	1.9010E+00	0.0000E+00	9182	7950	7	2	67.48
OBSTCLAL	0	15625	1.9010E+00	0.0000E+00	7952	7949	6	2	29.71
OBSTCLBL	0	15625	7.2958E+00	0.0000E+00	17171	11317	12	2	65.66
OBSTCLBM	0	15625	7.2958E+00	0.0000E+00	6977	11317	3	2	34.69
OBSTCLBU	0	15625	7.2958E+00	0.0000E+00	13094	11317	10	2	47.66
ODNAMUR	0	11130	9.2366E+03	1.0550E-15	4686	5512	2	2	663.00

Table 2: Results for SQIC with LUSOL on ''large'' CUTEr QPs (continued)

Name	m	n	Objective	$\|Ax-b\|$	Itn	nS	bkFac	nFac	Time
OSLBQP	0	8	6.2500E+00	0.0000E+00	7	6	0	1	0.00
PALMER1C	0	8	9.7605E-02	0.0000E+00	9	8	0	2	0.00
PALMER1D	0	7	6.5267E-01	2.0708E-16	8	7	0	1	0.00
PALMER2C	0	8	1.4369E-02	4.1276E-17	9	8	0	2	0.00
PALMER3C	0	8	1.9538E-02	8.0822E-17	9	8	0	2	0.00
PALMER4C	0	8	5.0311E-02	2.4247E-16	9	8	0	2	0.00
PRIMAL3	111	745	-1.3576E-01	6.5629E-13	711	571	0	3	0.30
PRIMAL4	75	1489	-7.4609E-01	5.2297E-16	1223	1140	0	2	1.18
RDW2D51U	65025	132098	8.3606E-04	2.3948E-06	2514	65025	1	5	559.73
RDW2D52U	65025	132098	1.1373E-02	1.9791E-06	2638	65025	1	32	647.47
SOSQP2	10001	20000	-4.9987E+03	1.6293E-07	15238	4982	2	48	74.59
STCQP1	4095	8193	3.6710E+05	2.7970E-15	3570	5717	1	19	9.09
STCQP2	4095	8193	3.7189E+04	3.1792E-16	5799	3970	1	67	9.70
STNQP1	4095	8193	-3.1170E+05	1.2952E-15	4008	5277	1	19	9.45
STNQP2	4095	8193	-5.7497E+05	4.2823E-16	6612	2640	1	67	10.16
TESTQUAD	0	5000	0.0000E+00	0.0000E+00	2003	5000	1	2	5.60
TOINTQOR	0	50	1.1755E+03	2.8758E-16	51	50	0	1	0.00
TORSION1	0	14884	-4.2570E-01	0.0000E+00	9987	9984	8	2	44.21
TORSION2	0	14884	-4.2570E-01	0.0000E+00	6460	9984	5	2	46.76
TORSION3	0	14884	-1.2122E+00	0.0000E+00	5211	5208	4	2	20.57
TORSION4	0	14884	-1.2122E+00	0.0000E+00	11399	5208	9	2	75.60
TORSION5	0	14884	-2.8588E+00	0.0000E+00	2571	2568	1	2	8.97
TORSION6	0	14884	-2.8588E+00	0.0000E+00	15017	2568	12	3	89.64
TORSIONA	0	14884	-4.1842E-01	0.0000E+00	10115	10112	8	2	45.07
TORSIONB	0	14884	-4.1842E-01	0.0000E+00	6292	10112	5	2	45.01
TORSIONC	0	14884	-1.2045E+00	0.0000E+00	5275	5272	4	2	20.59
TORSIOND	0	14884	-1.2045E+00	0.0000E+00	11298	5272	9	2	76.80
TORSIONE	0	14884	-2.8508E+00	0.0000E+00	2603	2600	1	2	9.04
TORSIONF	0	14884	-2.8508E+00	0.0000E+00	14909	2600	12	3	88.47
TRIDIA	0	10000	-8.8818E-16	0.0000E+00	2003	10000	1	2	11.81
UBH1	12000	18009	1.1160E+00	3.0585E-11	5880	5997	1	90	27.79
WALL10	0	1461	-4.5595E+05	8.6179E-16	1434	1101	0	16	2.62
WALL100	0	149624	-8.9544E+03	1.0332E-17	6113	110712	4	2	1265.84
WALL20	0	5924	-5.2210E+06	1.4350E-19	3034	4277	6	2	14.77
WALL50	0	37311	-9.5450E+06	2.5830E-18	3585	26961	40	2	525.42
ZANGWIL2	0	2	-1.8200E+01	9.7205E-16	3	2	0	1	0.00

Table 3: Results for SQIC with HSL_MA57 on ''small'' CUTEr QPs

Name	m	n	Objective	$\|Ax-b\|$	Itn	nS	bkFac	nFac	Time
A0ENDNDL	15002	45006	0.0000E+00	1.2049E-11	7276	0	0	152	26.06
A0ENINDL	15002	45006	0.0000E+00	3.4195E-12	7226	0	0	149	24.61
A0ENSNDL	15002	45006	-2.2319E-09	3.1831E-09	5781	0	0	120	20.73
A0ESDNDL	15002	45006	0.0000E+00	5.2419E-12	7192	0	0	140	26.23
A0ESINDL	15002	45006	0.0000E+00	2.1649E-12	7165	0	0	147	24.43
A0ESSNDL	15002	45006	-2.2519E-09	5.4170E-09	5865	0	0	121	22.72
A0NNDNDL	20004	60012	0.0000E+00	1.1623E-12	61929	0	0	1261	406.66
A0NNDNIL	20004	60012	4.8858E+01	4.4889E-12	12155	61	0	244	54.33
A0NNDNSL	20004	60012	-6.6875E-09	9.2523E-12	39329	0	0	795	211.03
A0NNSNSL	20004	60012	-1.1023E-08	6.3086E-12	21119	0	0	431	97.46
A0NSDSDL	20004	60012	0.0000E+00	8.8097E-12	30797	0	0	629	149.14
A0NSDSDS	2004	6012	-3.8600E-09	2.0915E-07	1615	0	0	37	0.60
A0NSDSIL	20004	60012	0.0000E+00	3.3788E-11	13753	0	0	281	60.43
A0NSDSSL	20004	60012	-9.3001E-11	1.2024E-11	23318	0	0	477	94.17
A0NSSSSL	20004	60012	-1.5102E-08	6.3385E-12	17173	0	0	352	67.33
A2ENDNDL	15002	45006	0.0000E+00	4.6408E-12	7117	0	0	143	26.35
A2ENINDL	15002	45006	5.6153E-26	3.6365E-12	7114	2	0	143	26.26
A2ENSNDL	15002	45006	-2.9789E-09	6.1114E-12	5568	1	0	115	21.04
A2ESDNDL	15002	45006	9.6115E-27	5.2652E-13	6975	8	0	132	26.34
A2ESINDL	15002	45006	1.4546E-27	3.7362E-13	7009	4	0	142	26.34

Table 3: Results for SQIC with HSL_MA57 on ‘‘small’’ CUTEr QPs (continued)

Name	m	n	Objective	$\|Ax-b\|$	Itn	nS	bkFac	nFac	Time
A2ESSNDL	15002	45006	-2.9789E-09	6.7244E-12	5422	2	0	112	21.12
A2NNDNDL	20004	60012	4.4271E-11	5.8288E-08	71795	21	0	1450	511.13
A2NNDNIL^i	20004	60012	0.0000E+00	6.4845E-14	11717	0	0	241	46.42
A2NNDNSL	20004	60012	0.0000E+00	2.7387E-12	44025	0	0	893	265.90
A2NNSNSL	20004	60012	-2.5420E-09	7.4080E-08	22765	0	0	465	86.71
A2NSDSDL	20004	60012	-2.7289E-12	6.3848E-08	50809	8	0	1055	321.89
A2NSDSIL	20004	60012	5.3679E+01	7.4628E-08	15050	253	0	301	86.73
A2NSDSSL	20004	60012	-3.2536E-12	7.4640E-12	29605	0	0	654	167.94
A2NSSSSL	20004	60012	-8.7014E-05	2.1586E-07	20498	0	0	419	77.68
A5ENDNDL	15002	45006	4.8867E-26	3.7520E-09	6606	18	0	128	25.29
A5ENINDL	15002	45006	-2.7579E-29	7.7338E-09	6613	3	0	130	25.02
A5ENSNDL	15002	45006	-1.1200E-09	8.3769E-12	5111	0	0	105	19.87
A5ESDNDL	15002	45006	2.7862E-26	1.0486E-13	6442	6	0	128	27.04
A5ESINDL	15002	45006	5.9148E-27	6.4776E-12	6430	3	0	130	26.49
A5ESSNDL	15002	45006	-1.1317E-09	7.6808E-12	4785	0	0	97	15.14
A5NNDNDL	20004	60012	3.8603E+01	1.6980E-12	64923	183	0	1348	465.96
A5NNDNIL^i	20004	60012	0.0000E+00	9.3408E-14	10259	0	0	211	42.18
A5NNDNSL	20004	60012	1.7860E-11	2.0308E-11	41977	0	0	867	246.34
A5NNSNSL	20004	60012	-2.2752E-09	2.6288E-07	25628	0	0	519	125.32
A5NSDSDL^r	20004	60012	1.0300E+04	7.1406E-13	47007	1	0	1025	384.28
A5NSDSDM	2004	6012	-3.8600E-09	2.0915E-07	1615	0	0	37	0.61
A5NSDSIL	20004	60012	1.1413E+01	2.5155E-07	11945	614	0	230	75.51
A5NSDSSL^r	20004	60012	4.4189E+04	1.2859E-11	29762	1	0	666	209.08
A5NSSNSM	2004	6012	-3.8600E-09	2.0915E-07	1615	0	0	37	0.65
A5NSSSSL	20004	60012	-7.8358E-09	3.2229E-07	22517	0	0	460	108.88
AVGASA	10	8	-4.6319E+00	8.4581E-12	11	3	0	1	0.00
AVGASB	10	8	-4.4832E+00	1.3672E-16	11	3	0	1	0.00
BIGGSC4	7	4	-2.4375E+01	9.8000E-12	12	1	0	1	0.00
BLOCKQP1	5001	10010	-4.9940E+03	1.8622E-16	5014	9	0	2	4.78
BLOCKQP2	5001	10010	-4.9928E+03	3.0600E-08	7516	9	0	104	13.25
BLOCKQP3	5001	10010	-2.4950E+03	1.8223E-16	5014	9	0	2	5.31
BLOCKQP4	5001	10010	-2.4933E+03	3.0600E-08	8493	9	0	104	18.59
BLOCKQP5	5001	10010	-2.4950E+03	1.7768E-16	5020	9	0	2	4.14
BQP1VAR	0	1	0.0000E+00	0.0000E+00	2	0	0	1	0.00
CVXBQP1	0	10000	2.2502E+06	1.8646E-17	10001	0	0	1	7.99
CVXQP3	7500	10000	1.1571E+08	1.7545E-09	11418	390	0	196	11.60
DEGENQP	8010	20	0.0000E+00	0.0000E+00	12	0	0	2	0.02
DUALC1	215	9	6.1553E+03	3.5126E-15	5	2	0	2	0.00
DUALC2	229	7	3.5513E+03	2.1456E-15	4	2	0	2	0.00
DUALC5	278	8	4.2723E+02	9.8919E-16	5	4	0	2	0.00
DUALC8	503	8	1.8309E+04	2.1421E-15	7	2	0	2	0.00
FERRISDC	210	2200	0.0000E+00	0.0000E+00	1	0	0	2	0.27
GENHS28	8	10	9.2717E-01	1.8625E-15	3	2	0	2	0.00
GMNCASE1	300	175	2.6697E-01	3.8875E-16	102	53	0	1	0.03
GMNCASE2	1050	175	-9.9444E-01	3.6566E-15	104	46	0	2	0.04
GMNCASE3	1050	175	1.5251E+00	1.2018E-14	107	48	0	2	0.04
GMNCASE4	350	175	5.9469E+03	7.7433E-12	141	0	0	3	0.06
GOULDQP1	17	32	-3.4853E+03	1.2250E-11	23	0	0	2	0.00
GOULDQP2	9999	19999	1.8512E-12	1.4803E-16	1	0	0	2	0.29
HARKERP2	0	1000	-5.0000E-01	0.0000E+00	1000	0	0	1	4.48
HATFLDH	7	4	-2.4500E+01	0.0000E+00	4	0	0	1	0.00
HS118	17	15	6.6482E+02	1.2250E-11	23	0	0	1	0.00
HS21	1	2	-9.9960E+01	1.7764E-15	2	1	0	1	0.00
HS268	5	5	-7.2760E-12	2.5794E-15	11	5	0	1	0.00
HS3	0	2	0.0000E+00	0.0000E+00	3	1	0	1	0.00
HS35	1	3	1.1111E-01	3.4648E-11	6	2	0	1	0.00
HS35I	1	3	1.1111E-01	3.4648E-11	6	2	0	1	0.00
HS35MOD	1	3	2.5000E-01	0.0000E+00	2	1	0	1	0.00
HS3MOD	0	2	2.2187E-31	0.0000E+00	3	1	0	1	0.00
HS44	6	4	-1.3000E+01	2.6585E-16	3	0	0	1	0.00

Table 3: Results for SQIC with HSL_MA57 on ‘‘small’’ CUTEr QPs (continued)

Name	m	n	Objective	$\|Ax-b\|$	Itn	nS	bkFac	nFac	Time
HS44NEW	6	4	-1.3000E+01	3.9878E-16	6	0	0	1	0.00
HS51	3	5	-8.8818E-16	1.5632E-16	3	2	0	2	0.00
HS52	3	5	5.3266E+00	1.0747E-16	3	2	0	2	0.00
HS53	3	5	4.0930E+00	1.5632E-16	3	2	0	2	0.00
HS76	3	4	-4.6818E+00	3.0304E-16	5	2	0	1	0.00
HS76I	3	4	-4.6818E+00	3.0304E-16	5	2	0	1	0.00
KSIP	1001	20	5.7580E-01	1.4470E-17	1536	18	0	65	0.86
LEUVEN1	2220	1530	-1.5243E+07	7.2852E-09	1516	12	0	31	0.42
LEUVEN2	2329	1530	-1.4147E+07	1.6956E-08	610	2	0	12	0.16
LEUVEN3	2973	1200	-1.0381E+09	6.0444E-09	1079	50	0	29	3.15
LEUVEN4	2973	1200	-1.4083E+09	2.1971E-11	1709	50	0	71	6.80
LEUVEN5	2973	1200	-1.0381E+09	6.0444E-09	1079	50	0	29	3.15
LEUVEN6	3091	1200	-1.4533E+08	1.6050E-09	578	30	0	26	2.57
LEUVEN7	946	360	6.9455E+02	2.4889E-12	203	19	0	4	0.12
LINCONT^i	419	1257	0.0000E+00	4.6663E-14	126	0	0	4	0.03
LISWET1	10000	10002	3.6121E+01	1.3323E-15	4	2	0	2	0.15
LISWET10	10000	10002	4.9483E+01	1.1435E-14	49	16	0	4	0.24
LISWET11	10000	10002	4.9524E+01	4.9220E-15	43	30	0	4	0.25
LISWET12	10000	10002	1.7369E+03	9.2519E-16	28	5	0	4	0.21
LISWET2	10000	10002	2.5000E+01	1.2212E-15	22	4	0	2	0.18
LISWET3	10000	10002	2.5000E+01	2.8630E-08	444	261	0	5	0.86
LISWET4	10000	10002	2.5000E+01	2.0362E-07	431	270	0	5	0.85
LISWET5	10000	10002	2.5000E+01	4.9305E-08	412	254	0	5	0.78
LISWET6	10000	10002	2.5000E+01	2.3685E-15	341	222	0	5	0.66
LISWET7	10000	10002	4.9884E+02	7.7716E-16	4	2	0	2	0.15
LISWET8	10000	10002	7.1447E+02	6.6613E-16	22	13	0	2	0.19
LISWET9	10000	10002	1.9632E+03	1.7023E-15	16	4	0	4	0.21
LOTSCHD	7	12	2.3984E+03	3.4710E-15	8	0	0	2	0.00
MARATOSB^u	0	2	-1.4400E+06	2.0016E-17	2	0	0	1	0.00
MPC1	3833	2550	-2.3262E+07	1.8844E-08	1362	0	0	26	0.52
MPC10	2351	1530	-1.5034E+07	4.8004E-08	1180	11	0	24	0.35
MPC11	2351	1530	-1.5030E+07	9.1770E-09	924	34	0	17	0.27
MPC12	2351	1530	-1.5033E+07	1.9375E-07	1155	19	0	22	0.35
MPC13	2351	1530	-1.5034E+07	4.9520E-09	1070	13	0	21	0.31
MPC14	2351	1530	-1.5034E+07	3.1434E-07	1190	16	0	22	0.34
MPC15	2351	1530	-1.5034E+07	2.1451E-06	997	15	0	19	0.29
MPC16	2351	1530	-1.5034E+07	8.5862E-07	1047	16	0	20	0.30
MPC2	2351	1530	-1.5033E+07	9.2528E-08	1201	27	0	21	0.35
MPC3	2351	1530	-1.5030E+07	2.3069E-08	1228	32	0	23	0.39
MPC4	2351	1530	-1.5033E+07	4.7017E-09	1259	21	0	24	0.36
MPC5	2351	1530	-1.5033E+07	9.2840E-08	1271	25	0	23	0.35
MPC6	2351	1530	-1.5034E+07	4.7094E-09	1179	18	0	22	0.35
MPC7	2351	1530	-1.5034E+07	2.3473E-09	1087	15	0	21	0.31
MPC8	2351	1530	-1.5034E+07	5.1577E-09	1158	13	0	22	0.35
MPC9	2351	1530	-1.5034E+07	1.9332E-07	1185	11	0	23	0.35
NASH^i	24	72	0.0000E+00	9.4369E-16	2	0	0	2	0.00
NCVXBQP1	0	10000	-1.9855E+10	0.0000E+00	10015	0	0	1	8.23
NCVXBQP2	0	10000	-1.3245E+10	1.8604E-17	11223	51	0	1	9.24
NCVXBQP3	0	10000	-6.4122E+09	2.3813E-15	10837	126	0	1	8.74
NCVXQP1	500	1000	-7.1562E+07	6.8979E-11	749	0	0	8	0.09
NCVXQP2	500	1000	-5.7759E+07	4.2449E-11	1057	0	0	16	0.15
NCVXQP3	500	1000	-2.9253E+07	3.6206E-11	1223	19	0	13	0.20
NCVXQP4	250	1000	-9.3995E+07	1.0037E-11	788	0	0	4	0.09
NCVXQP5	250	1000	-6.6257E+07	1.0746E-11	824	0	0	5	0.09
NCVXQP6	250	1000	-3.4172E+07	6.9472E-12	926	49	0	6	0.12
NCVXQP7	750	1000	-4.3521E+07	7.4509E-11	660	0	0	11	0.08
NCVXQP8	750	1000	-3.0103E+07	9.6553E-11	909	0	0	16	0.12
NCVXQP9	750	1000	-2.1230E+07	2.6031E-09	961	11	0	15	0.16
PENTDI	0	5000	-7.5000E-01	0.0000E+00	3	2	0	1	0.07
PORTSNQP	2	100000	3.3332E+04	3.5313E-14	108265	257	0	882	42.93

Table 3: Results for SQIC with HSL_MA57 on ‘‘small’’ CUTEr QPs (continued)

Name	m	n	Objective	$\|Ax-b\|$	Itn	nS	bkFac	nFac	Time
PORTSQP	1	100000	3.3331E+04	1.8822E-11	100317	315	0	2	23.90
POWELL20	10000	10000	5.2090E+10	1.6977E-11	5003	1	0	103	6.08
PRIMAL1	85	325	-3.5013E-02	1.5669E-12	218	133	0	2	0.02
PRIMAL2	96	649	-3.3734E-02	3.0138E-16	408	302	0	2	0.09
PRIMALC1	9	230	-6.1553E+03	1.4648E-12	19	14	0	1	0.00
PRIMALC2	7	231	-3.5513E+03	1.1415E-13	4	1	0	1	0.00
PRIMALC5	8	287	-4.2723E+02	3.1580E-14	10	5	0	1	0.00
PRIMALC8	8	520	-1.8309E+04	1.5526E-10	22	17	0	1	0.00
QPBAND	5000	10000	-9.9992E+03	1.8821E-11	29960	39	0	204	23.44
QPCBLEND	74	83	-7.8425E-03	1.5447E-09	76	2	0	4	0.00
QPCBOEI1	351	384	1.1504E+07	1.2362E-10	701	113	0	9	0.05
QPCBOEI2	166	143	8.1720E+06	5.0579E-11	203	32	0	4	0.01
QPCSTAIR	356	467	6.2044E+06	2.8087E-12	304	20	0	8	0.03
QPNBAND	5000	10000	-4.9997E+04	7.2371E-16	15000	1	0	103	12.11
QPNBLEND	74	83	-8.7056E-03	1.8434E-11	72	3	0	2	0.00
QPNBOEI1	351	384	6.7367E+06	1.9434E-10	685	92	0	11	0.05
QPNBOEI2	166	143	1.3683E+06	5.7753E-11	229	27	0	5	0.01
QPNSTAIR	356	467	5.1460E+06	1.1941E-11	349	20	0	6	0.03
QUDLIN	0	5000	-1.2500E+09	0.0000E+00	5000	0	0	1	0.84
RDW2D51F	65025	132098	1.1353E-03	2.9003E-07	2258	0	0	33	1492.58
RDW2D52F	49	162	8.6159E-03	1.3853E-15	71	37	0	3	0.00
S268	5	5	-7.2760E-12	2.5794E-15	11	5	0	1	0.00
SIM2BQP	0	2	0.0000E+00	0.0000E+00	2	0	0	1	0.00
SIMBQP	0	2	3.4667E-31	0.0000E+00	3	1	0	1	0.00
SOSQP1	10001	20000	-2.4500E-11	1.5765E-11	3	0	0	2	0.67
STATIC3^u	96	434	-3.0892E+02	0.0000E+00	3	1	0	2	0.00
STEENBRA	108	432	1.6958E+04	8.1667E-11	87	11	0	3	0.01
TAME	1	2	0.0000E+00	6.0309E-17	2	1	0	2	0.00
YAO	2000	2002	1.9770E+02	9.2519E-17	3	1	0	1	0.01
ZECEVIC2	2	2	-4.1250E+00	0.0000E+00	3	1	0	1	0.00

Table 4: Results for SQIC with HSL_MA57 on ‘‘large’’ CUTEr QPs

Name	m	n	Objective	$\|Ax-b\|$	Itn	nS	bkFac	nFac	Time
ALLINQP	25000	50000	-5.4813E+03	4.1613E-15	18292	9820	2	75	139.24
AUG2D	10000	20200	1.6874E+06	1.3263E-12	2003	10200	1	3	32.81
AUG2DC	10000	20200	1.8184E+06	8.3370E-13	2003	10200	1	3	12.46
AUG2DCQP	10000	20200	6.4981E+06	1.4282E-12	14460	9994	8	83	101.10
AUG2DQP	10000	20200	6.2370E+06	1.6333E-11	14440	9801	8	78	101.50
AUG3D	8000	27543	2.4561E+04	2.6534E-11	16912	16909	16	3	298.81
AUG3DC	8000	27543	2.7654E+04	8.7782E-14	2003	19543	1	3	14.26
AUG3DCQP	8000	27543	6.1560E+04	7.3399E-08	22204	17665	16	82	227.38
AUG3DQP	8000	27543	5.4229E+04	1.6334E-11	18543	13712	12	83	168.58
BIGGSB1	0	5000	1.5000E-02	0.0000E+00	5005	4998	3	2	13.68
BLOWEYA	2002	4002	-2.2781E-02	7.3505E-14	2002	2000	0	2	6.00
BLOWEYB	2002	4002	-1.5226E-02	9.3323E-14	2002	2000	0	2	6.16
BLOWEYC	2002	4002	-1.5246E-02	1.5728E-12	2003	2000	0	2	6.00
BQPGABIM	0	50	-3.7903E-05	0.0000E+00	41	36	0	1	0.00
BQPGASIM	0	50	-5.5198E-05	0.0000E+00	45	40	0	1	0.00
BQPGAUSS	0	2003	-3.6258E-01	8.4349E-17	2236	1909	0	1	6.38
CHENHARK	0	5000	-2.0000E+00	0.0000E+00	6003	3000	1	2	7.23
CVXQP1	5000	10000	1.0870E+08	6.9982E-11	11491	1261	0	121	16.84
CVXQP2	2500	10000	8.1842E+07	2.4684E-13	8298	2210	1	25	24.51
DIXON3DQ	0	10000	-4.4409E-16	0.0000E+00	2003	10000	1	2	7.56
DQDRTIC	0	5000	0.0000E+00	0.0000E+00	2003	5000	1	2	5.58
DTOC3	9998	14999	2.3526E+02	2.3683E-15	2004	4999	1	3	12.80
DUAL1	1	85	3.5013E-02	1.3715E-16	77	62	0	2	0.01
DUAL2	1	96	3.3734E-02	1.5379E-16	94	91	0	2	0.01
DUAL3	1	111	1.3576E-01	2.8691E-16	111	96	0	2	0.01
DUAL4	1	75	7.4609E-01	2.2421E-16	64	61	0	2	0.00

Table 4: Results for SQIC with HSL_MA57 on ''large'' CUTEr QPs (continued)

Name	m	n	Objective	$\|Ax-b\|$	Itn	nS	bkFac	nFac	Time
GOULDQP3	9999	19999	2.3796E-05	1.2512E-15	5899	4988	3	19	112.99
GRIDNETA	6724	13284	3.0498E+02	7.2053E-14	2083	2218	1	4	8.27
GRIDNETB	6724	13284	1.4332E+02	1.4862E-13	2003	6561	1	3	8.47
GRIDNETC	6724	13284	1.4832E+02	4.4210E-14	3930	4533	2	4	19.29
HILBERTA	0	10	9.6007E-09	0.0000E+00	8	7	0	1	0.00
HILBERTB	0	50	5.2590E-28	9.5890E-16	51	50	0	1	0.00
HUES-MOD	2	10000	3.4824E+07	1.4314E-11	3828	9444	1	3	8.95
HUESTIS	2	10000	3.4824E+11	1.2023E-12	3827	9444	1	3	9.02
JNLBRNG1	0	15625	-1.8058E-01	0.0000E+00	4020	10247	2	2	19.57
JNLBRNG2	0	15625	-4.1496E+00	0.0000E+00	3639	9139	2	2	16.50
JNLBRNGA	0	15625	-2.6851E-01	0.0000E+00	9971	9968	8	2	44.65
JNLBRNGB	0	15625	-6.2807E+00	0.0000E+00	8480	8477	7	2	37.00
MOSARQP1	700	2500	-3.8214E+03	6.6779E-11	3254	1021	0	23	3.68
MOSARQP2	700	2500	-5.0526E+03	4.4410E-14	2553	1640	0	6	4.01
NOBNDTOR	0	14884	-4.4054E-01	0.0000E+00	5899	12078	4	2	31.47
OBSTCLAE	0	15625	1.9010E+00	0.0000E+00	9182	7950	7	2	69.47
OBSTCLAL	0	15625	1.9010E+00	0.0000E+00	7952	7949	6	2	33.69
OBSTCLBL	0	15625	7.2958E+00	0.0000E+00	17171	11317	12	2	79.54
OBSTCLBM	0	15625	7.2958E+00	0.0000E+00	6977	11317	3	2	36.26
OBSTCLBU	0	15625	7.2958E+00	0.0000E+00	13094	11317	10	2	55.56
ODNAMUR	0	11130	9.2366E+03	0.0000E+00	4686	5512	2	2	623.01
OSLBQP	0	8	6.2500E+00	0.0000E+00	7	6	0	1	0.00
PALMER1C	0	8	9.7605E-02	0.0000E+00	9	8	0	2	0.00
PALMER1D	0	7	6.5267E-01	2.0708E-16	8	7	0	1	0.00
PALMER2C	0	8	1.4369E-02	4.1276E-17	9	8	0	2	0.00
PALMER3C	0	8	1.9538E-02	8.0822E-17	9	8	0	2	0.00
PALMER4C	0	8	5.0311E-02	2.4247E-16	9	8	0	2	0.00
PRIMAL3	111	745	-1.3576E-01	6.5629E-13	711	571	0	3	0.30
PRIMAL4	75	1489	-7.4609E-01	5.2297E-16	1223	1140	0	2	1.18
RDW2D51U	65025	132098	8.3606E-04	8.3439E-09	2514	65025	1	5	336.86
RDW2D52U	65025	132098	1.1373E-02	1.9130E-08	2638	65025	1	32	423.98
SOSQP2	10001	20000	-4.9987E+03	1.6293E-07	15238	4982	2	48	248.08
STCQP1	4095	8193	3.6710E+05	9.1112E-14	3570	5717	1	19	9.37
STCQP2	4095	8193	3.7189E+04	3.4885E-15	5799	3970	1	67	10.27
STNQP1	4095	8193	-3.1170E+05	7.2110E-12	4008	5277	1	19	9.70
STNQP2	4095	8193	-5.7497E+05	3.7002E-14	6612	2640	1	67	10.19
TESTQUAD	0	5000	0.0000E+00	0.0000E+00	2003	5000	1	2	5.57
TOINTQOR	0	50	1.1755E+03	2.8758E-16	51	50	0	1	0.00
TORSION1	0	14884	-4.2570E-01	0.0000E+00	9987	9984	8	2	49.54
TORSION2	0	14884	-4.2570E-01	0.0000E+00	6460	9984	5	2	47.72
TORSION3	0	14884	-1.2122E+00	0.0000E+00	5211	5208	4	2	22.05
TORSION4	0	14884	-1.2122E+00	0.0000E+00	11399	5208	9	2	80.15
TORSION5	0	14884	-2.8588E+00	0.0000E+00	2571	2568	1	2	9.20
TORSION6	0	14884	-2.8588E+00	0.0000E+00	15017	2568	12	3	94.57
TORSIONA	0	14884	-4.1842E-01	0.0000E+00	10115	10112	8	2	50.22
TORSIONB	0	14884	-4.1842E-01	0.0000E+00	6292	10112	5	2	47.65
TORSIONC	0	14884	-1.2045E+00	0.0000E+00	5275	5272	4	2	22.18
TORSIOND	0	14884	-1.2045E+00	0.0000E+00	11298	5272	9	2	79.47
TORSIONE	0	14884	-2.8508E+00	0.0000E+00	2603	2600	1	2	9.21
TORSIONF	0	14884	-2.8508E+00	0.0000E+00	14909	2600	12	3	94.08
TRIDIA	0	10000	-1.1102E-15	5.5728E-17	2003	10000	1	2	11.58
UBH1	12000	18009	1.1160E+00	1.0281E-11	5880	5997	1	90	27.64
WALL10	0	1461	-4.5595E+05	8.6179E-16	1434	1101	0	16	2.62
WALL100	0	149624	-8.9544E+03	1.0906E-17	6113	110712	4	2	1225.79
WALL20	0	5924	-5.2210E+06	1.4350E-19	3034	4277	2	2	13.36
WALL50	0	37311	-9.5450E+06	2.5830E-18	3585	26961	2	2	75.91
ZANGWIL2	0	2	-1.8200E+01	9.7205E-16	3	2	0	1	0.00

Table 5: Results for SQIC with UMFPACK on ''small'' CUTEr QPs

Name	m	n	Objective	$\|Ax-b\|$	Itn	nS	bkFac	nFac	Time
A0ENDNDL	15002	45006	0.0000E+00	1.2049E-11	7276	0	0	152	25.48
A0ENINDL	15002	45006	0.0000E+00	3.4195E-12	7226	0	0	149	24.60
A0ENSNDL	15002	45006	-2.2319E-09	3.1831E-09	5781	0	0	120	20.70
A0ESDNDL	15002	45006	0.0000E+00	5.2419E-12	7192	0	0	140	25.81
A0ESINDL	15002	45006	0.0000E+00	2.1649E-12	7165	0	0	147	24.47
A0ESSNDL	15002	45006	-2.2519E-09	5.4170E-09	5865	0	0	121	22.78
A0NNDNDL	20004	60012	0.0000E+00	1.1623E-12	61929	0	0	1261	406.05
A0NNDNIL	20004	60012	4.8858E+01	4.4889E-12	12155	61	0	244	54.11
A0NNDNSL	20004	60012	-6.6875E-09	9.2523E-12	39329	0	0	795	209.99
A0NNSNSL	20004	60012	-1.1023E-08	6.3086E-12	21119	0	0	431	97.70
A0NSDSDL	20004	60012	0.0000E+00	8.8097E-12	30797	0	0	629	149.16
A0NSDSDS	2004	6012	-3.8600E-09	2.0915E-07	1615	0	0	37	0.64
A0NSDSIL	20004	60012	0.0000E+00	3.3788E-11	13753	0	0	281	60.11
A0NSDSSL	20004	60012	-9.3001E-11	1.2024E-11	23318	0	0	477	94.07
A0NSSSSL	20004	60012	-1.5102E-08	6.3385E-12	17173	0	0	352	67.29
A2ENDNDL	15002	45006	0.0000E+00	4.6408E-12	7117	0	0	143	26.43
A2ENINDL	15002	45006	5.6153E-26	3.6365E-12	7114	2	0	143	26.21
A2ENSNDL	15002	45006	-2.9789E-09	6.1114E-12	5568	1	0	115	21.08
A2ESDNDL	15002	45006	9.6115E-27	5.2652E-13	6975	8	0	132	26.41
A2ESINDL	15002	45006	1.4546E-27	3.7362E-13	7009	4	0	142	26.37
A2ESSNDL	15002	45006	-2.9789E-09	6.7244E-12	5422	2	0	112	21.26
A2NNDNDL	20004	60012	4.4271E-11	5.8288E-08	71795	21	0	1450	512.07
A2NNDNIL^{i}	20004	60012	0.0000E+00	6.4845E-14	11717	0	0	241	48.06
A2NNDNSL	20004	60012	0.0000E+00	2.7387E-12	44025	0	0	893	266.52
A2NNSNSL	20004	60012	-2.5420E-09	7.4080E-08	22765	0	0	465	86.35
A2NSDSDL	20004	60012	-2.7289E-12	6.3848E-08	50809	8	0	1055	321.94
A2NSDSIL	20004	60012	5.3679E+01	7.4628E-08	15050	253	0	301	86.75
A2NSDSSL	20004	60012	-3.2536E-12	7.4640E-12	29605	0	0	654	167.34
A2NSSSSL	20004	60012	-8.7014E-05	2.1586E-07	20498	0	0	419	78.25
A5ENDNDL	15002	45006	4.8867E-26	3.7520E-09	6606	18	0	128	25.20
A5ENINDL	15002	45006	-2.7579E-29	7.7338E-09	6613	3	0	130	25.30
A5ENSNDL	15002	45006	-1.1200E-09	8.3769E-12	5111	0	0	105	19.76
A5ESDNDL	15002	45006	2.7862E-26	1.0486E-13	6442	6	0	128	27.19
A5ESINDL	15002	45006	5.9148E-27	6.4776E-12	6430	3	0	130	26.22
A5ESSNDL	15002	45006	-1.1317E-09	7.6808E-12	4785	0	0	97	15.57
A5NNDNDL	20004	60012	3.8603E+01	1.6980E-12	64923	183	0	1348	465.04
A5NNDNIL^{i}	20004	60012	0.0000E+00	9.3408E-14	10259	0	0	211	41.38
A5NNDNSL	20004	60012	1.7860E-11	2.0308E-11	41977	0	0	867	245.45
A5NNSNSL	20004	60012	-2.2752E-09	2.6288E-07	25628	0	0	519	125.85
A5NSDSDL^{r}	20004	60012	1.0300E+04	7.1406E-13	47007	1	0	1025	384.24
A5NSDSDM	2004	6012	-3.8600E-09	2.0915E-07	1615	0	0	37	0.64
A5NSDSIL	20004	60012	1.1413E+01	2.5155E-07	11945	614	0	230	75.96
A5NSDSSL^{r}	20004	60012	4.4189E+04	1.2859E-11	29762	1	0	666	208.66
A5NSSNSM	2004	6012	-3.8600E-09	2.0915E-07	1615	0	0	37	0.59
A5NSSSSL	20004	60012	-7.8358E-09	3.2229E-07	22517	0	0	460	108.55
AVGASA	10	8	-4.6319E+00	8.4581E-12	11	3	0	1	0.00
AVGASB	10	8	-4.4832E+00	1.3672E-16	11	3	0	1	0.00
BIGGSC4	7	4	-2.4375E+01	9.8000E-12	12	1	0	1	0.00
BLOCKQP1	5001	10010	-4.9940E+03	1.8622E-16	5014	9	0	2	4.96
BLOCKQP2	5001	10010	-4.9928E+03	3.0600E-08	7516	9	0	104	13.60
BLOCKQP3	5001	10010	-2.4950E+03	1.8223E-16	5014	9	0	2	5.32
BLOCKQP4	5001	10010	-2.4933E+03	3.0600E-08	8493	9	0	104	18.73
BLOCKQP5	5001	10010	-2.4950E+03	1.7768E-16	5020	9	0	2	4.14
BQP1VAR	0	1	0.0000E+00	0.0000E+00	2	0	0	1	0.00
CVXBQP1	0	10000	2.2502E+06	1.8646E-17	10001	0	0	1	7.95
CVXQP3	7500	10000	1.1571E+08	1.7545E-09	11418	390	0	196	11.62
DEGENQP	8010	20	0.0000E+00	0.0000E+00	12	0	0	2	0.02
DUALC1	215	9	6.1553E+03	3.5126E-15	5	2	0	2	0.00
DUALC2	229	7	3.5513E+03	2.1456E-15	4	2	0	2	0.00
DUALC5	278	8	4.2723E+02	9.8919E-16	5	4	0	2	0.00
DUALC8	503	8	1.8309E+04	2.1421E-15	7	2	0	2	0.00

Table 5: Results for SQIC with UMFPACK on ''small'' CUTEr QPs (continued)

Name	m	n	Objective	$\|Ax-b\|$	Itn	nS	bkFac	nFac	Time
FERRISDC	210	2200	0.0000E+00	0.0000E+00	1	0	0	2	0.26
GENHS28	8	10	9.2717E-01	1.8625E-15	3	2	0	2	0.00
GMNCASE1	300	175	2.6697E-01	3.8875E-16	102	53	0	1	0.03
GMNCASE2	1050	175	-9.9444E-01	3.6566E-15	104	46	0	2	0.04
GMNCASE3	1050	175	1.5251E+00	1.2018E-14	107	48	0	2	0.04
GMNCASE4	350	175	5.9469E+03	7.7433E-12	141	0	0	3	0.05
GOULDQP1	17	32	-3.4853E+03	1.2250E-11	23	0	0	2	0.00
GOULDQP2	9999	19999	1.8512E-12	1.4803E-16	1	0	0	2	0.36
HARKERP2	0	1000	-5.0000E-01	0.0000E+00	1000	0	0	1	4.48
HATFLDH	7	4	-2.4500E+01	0.0000E+00	4	0	0	1	0.00
HS118	17	15	6.6482E+02	1.2250E-11	23	0	0	1	0.00
HS21	1	2	-9.9960E+01	1.7764E-15	2	1	0	1	0.00
HS268	5	5	-7.2760E-12	2.5794E-15	11	5	0	1	0.00
HS3	0	2	0.0000E+00	0.0000E+00	3	1	0	1	0.00
HS35	1	3	1.1111E-01	3.4648E-11	6	2	0	1	0.00
HS35I	1	3	1.1111E-01	3.4648E-11	6	2	0	1	0.00
HS35MOD	1	3	2.5000E-01	0.0000E+00	2	1	0	1	0.00
HS3MOD	0	2	2.2187E-31	0.0000E+00	3	1	0	1	0.00
HS44	6	4	-1.3000E+01	2.6585E-16	3	0	0	1	0.00
HS44NEW	6	4	-1.3000E+01	3.9878E-16	6	0	0	1	0.00
HS51	3	5	-8.8818E-16	1.5632E-16	3	2	0	2	0.00
HS52	3	5	5.3266E+00	1.0747E-16	3	2	0	2	0.00
HS53	3	5	4.0930E+00	1.5632E-16	3	2	0	2	0.00
HS76	3	4	-4.6818E+00	3.0304E-16	5	2	0	1	0.00
HS76I	3	4	-4.6818E+00	3.0304E-16	5	2	0	1	0.00
KSIP	1001	20	5.7580E-01	1.4470E-17	1536	18	0	65	0.86
LEUVEN1	2220	1530	-1.5243E+07	7.2852E-09	1516	12	0	31	0.42
LEUVEN2	2329	1530	-1.4147E+07	1.6956E-08	610	2	0	12	0.16
LEUVEN3	2973	1200	-1.0381E+09	6.0444E-09	1079	50	0	29	3.15
LEUVEN4	2973	1200	-1.4083E+09	2.1971E-11	1709	50	0	71	6.80
LEUVEN5	2973	1200	-1.0381E+09	6.0444E-09	1079	50	0	29	3.15
LEUVEN6	3091	1200	-1.4533E+08	1.6050E-09	578	30	0	26	2.57
LEUVEN7	946	360	6.9455E+02	2.4889E-12	203	19	0	4	0.12
LINCONT^i	419	1257	0.0000E+00	4.6663E-14	126	0	0	4	0.03
LISWET1	10000	10002	3.6121E+01	1.3323E-15	4	2	0	2	0.15
LISWET10	10000	10002	4.9483E+01	1.1435E-14	49	16	0	4	0.24
LISWET11	10000	10002	4.9524E+01	4.9220E-15	43	30	0	4	0.25
LISWET12	10000	10002	1.7369E+03	9.2519E-16	28	5	0	4	0.21
LISWET2	10000	10002	2.5000E+01	1.2212E-15	22	4	0	2	0.18
LISWET3	10000	10002	2.5000E+01	2.8630E-08	444	261	0	5	0.85
LISWET4	10000	10002	2.5000E+01	2.0362E-07	431	270	0	5	0.85
LISWET5	10000	10002	2.5000E+01	4.9305E-08	412	254	0	5	0.78
LISWET6	10000	10002	2.5000E+01	2.3685E-15	341	222	0	5	0.66
LISWET7	10000	10002	4.9884E+02	7.7716E-16	4	2	0	2	0.15
LISWET8	10000	10002	7.1447E+02	6.6613E-16	22	13	0	2	0.19
LISWET9	10000	10002	1.9632E+03	1.7023E-15	16	4	0	4	0.20
LOTSCHD	7	12	2.3984E+03	3.4710E-15	8	0	0	2	0.00
MARATOSB^u	0	2	-1.4400E+06	2.0016E-17	2	0	0	1	0.00
MPC1	3833	2550	-2.3262E+07	1.8844E-08	1362	0	0	26	0.52
MPC10	2351	1530	-1.5034E+07	4.8004E-08	1180	11	0	24	0.35
MPC11	2351	1530	-1.5030E+07	9.1770E-09	924	34	0	17	0.27
MPC12	2351	1530	-1.5033E+07	1.9375E-07	1155	19	0	22	0.35
MPC13	2351	1530	-1.5034E+07	4.9520E-09	1070	13	0	21	0.31
MPC14	2351	1530	-1.5034E+07	3.1434E-07	1190	16	0	22	0.34
MPC15	2351	1530	-1.5034E+07	2.1451E-06	997	15	0	19	0.29
MPC16	2351	1530	-1.5034E+07	8.5862E-07	1047	16	0	20	0.30
MPC2	2351	1530	-1.5033E+07	9.2528E-08	1201	27	0	21	0.35
MPC3	2351	1530	-1.5030E+07	2.3069E-08	1228	32	0	23	0.39
MPC4	2351	1530	-1.5033E+07	4.7017E-09	1259	21	0	24	0.36
MPC5	2351	1530	-1.5033E+07	9.2840E-08	1271	25	0	23	0.34

Table 5: Results for SQIC with UMFPACK on ‘‘small’’ CUTEr QPs (continued)

Name	m	n	Objective	$\|Ax-b\|$	Itn	nS	bkFac	nFac	Time
MPC6	2351	1530	-1.5034E+07	4.7094E-09	1179	18	0	22	0.35
MPC7	2351	1530	-1.5034E+07	2.3473E-09	1087	15	0	21	0.31
MPC8	2351	1530	-1.5034E+07	5.1577E-09	1158	13	0	22	0.35
MPC9	2351	1530	-1.5034E+07	1.9332E-07	1185	11	0	23	0.35
NASH^i	24	72	0.0000E+00	9.4369E-16	2	0	0	2	0.00
NCVXBQP1	0	10000	-1.9855E+10	0.0000E+00	10015	0	0	1	8.24
NCVXBQP2	0	10000	-1.3245E+10	1.8604E-17	11223	51	0	1	9.24
NCVXBQP3	0	10000	-6.4122E+09	2.3813E-15	10837	126	0	1	8.83
NCVXQP1	500	1000	-7.1562E+07	6.8979E-11	749	0	0	8	0.09
NCVXQP2	500	1000	-5.7759E+07	4.2449E-11	1057	0	0	16	0.15
NCVXQP3	500	1000	-2.9253E+07	3.6206E-11	1223	19	0	13	0.20
NCVXQP4	250	1000	-9.3995E+07	1.0037E-11	788	0	0	4	0.09
NCVXQP5	250	1000	-6.6257E+07	1.0746E-11	824	0	0	5	0.09
NCVXQP6	250	1000	-3.4172E+07	6.9472E-12	926	49	0	6	0.12
NCVXQP7	750	1000	-4.3521E+07	7.4509E-11	660	0	0	11	0.08
NCVXQP8	750	1000	-3.0103E+07	9.6553E-11	909	0	0	16	0.12
NCVXQP9	750	1000	-2.1230E+07	2.6031E-09	961	11	0	15	0.16
PENTDI	0	5000	-7.5000E-01	0.0000E+00	3	2	0	1	0.07
PORTSNQP	2	100000	3.3332E+04	3.5313E-14	108265	257	0	882	42.92
PORTSQP	1	100000	3.3331E+04	1.8822E-11	100317	315	0	2	23.91
POWELL20	10000	10000	5.2090E+10	1.6977E-11	5003	1	0	103	6.08
PRIMAL1	85	325	-3.5013E-02	1.5669E-12	218	133	0	2	0.03
PRIMAL2	96	649	-3.3734E-02	3.0138E-16	408	302	0	2	0.09
PRIMALC1	9	230	-6.1553E+03	1.4648E-12	19	14	0	1	0.00
PRIMALC2	7	231	-3.5513E+03	1.1415E-13	4	1	0	1	0.00
PRIMALC5	8	287	-4.2723E+02	3.1580E-14	10	5	0	1	0.00
PRIMALC8	8	520	-1.8309E+04	1.5526E-10	22	17	0	1	0.00
QPBAND	5000	10000	-9.9992E+03	1.8821E-11	29960	39	0	204	23.50
QPCBLEND	74	83	-7.8425E-03	1.5447E-09	76	2	0	4	0.00
QPCBOEI1	351	384	1.1504E+07	1.2362E-10	701	113	0	9	0.05
QPCBOEI2	166	143	8.1720E+06	5.0579E-11	203	32	0	4	0.01
QPCSTAIR	356	467	6.2044E+06	2.8087E-12	304	20	0	8	0.03
QPNBAND	5000	10000	-4.9997E+04	7.2371E-16	15000	1	0	103	12.16
QPNBLEND	74	83	-8.7056E-03	1.8434E-11	72	3	0	2	0.00
QPNBOEI1	351	384	6.7367E+06	1.9434E-10	685	92	0	11	0.05
QPNBOEI2	166	143	1.3683E+06	5.7753E-11	229	27	0	5	0.01
QPNSTAIR	356	467	5.1460E+06	1.1941E-11	349	20	0	6	0.03
QUDLIN	0	5000	-1.2500E+09	0.0000E+00	5000	0	0	1	0.85
RDW2D51F	65025	132098	1.1353E-03	2.9003E-07	2258	0	0	33	1495.16
RDW2D52F	49	162	8.6159E-03	1.3853E-15	71	37	0	3	0.00
S268	5	5	-7.2760E-12	2.5794E-15	11	5	0	1	0.00
SIM2BQP	0	2	0.0000E+00	0.0000E+00	2	0	0	1	0.00
SIMBQP	0	2	3.4667E-31	0.0000E+00	3	1	0	1	0.00
SOSQP1	10001	20000	-2.4500E-11	1.5765E-11	3	0	0	2	0.67
STATIC3^u	96	434	-3.0892E+02	0.0000E+00	3	1	0	2	0.00
STEENBRA	108	432	1.6958E+04	8.1667E-11	87	11	0	3	0.01
TAME	1	2	0.0000E+00	6.0309E-17	2	1	0	2	0.00
YAO	2000	2002	1.9770E+02	9.2519E-17	3	1	0	1	0.01
ZECEVIC2	2	2	-4.1250E+00	0.0000E+00	3	1	0	1	0.00

Table 6: Results for SQIC with UMFPACK on ‘‘large’’ CUTEr QPs

Name	m	n	Objective	$\|Ax-b\|$	Itn	nS	bkFac	nFac	Time
ALLINQP	25000	50000	-5.4813E+03	3.5582E-15	18292	9820	2	75	131.08
AUG2D	10000	20200	1.6874E+06	3.9317E-13	10195	10192	9	3	77.07
AUG2DC	10000	20200	1.8184E+06	1.1742E-12	2003	10200	1	3	13.22
AUG2DCQP	10000	20200	6.4981E+06	1.2138E-12	14456	9994	8	83	88.84
AUG2DQP	10000	20200	6.2370E+06	1.6333E-11	14438	9801	8	78	89.65
AUG3D	8000	27543	2.4561E+04	2.3625E-11	16912	16909	16	3	156.92

Table 6: Results for SQIC with UMFPACK on ''large''CUTEr QPs (continued)

Name	m	n	Objective	$\|Ax-b\|$	Itn	nS	bkFac	nFac	Time
AUG3DC	8000	27543	2.7654E+04	8.3341E-14	2003	19543	1	3	15.28
AUG3DCQP	8000	27543	6.1560E+04	3.6517E-08	22202	17665	16	82	190.52
AUG3DQP	8000	27543	5.4229E+04	1.6334E-11	18543	13712	12	83	138.60
BIGGSB1	0	5000	1.5000E-02	0.0000E+00	5005	4998	3	2	12.89
BLOWEYA	2002	4002	-2.2781E-02	7.3505E-14	2002	2000	0	2	5.93
BLOWEYB	2002	4002	-1.5226E-02	9.3323E-14	2002	2000	0	2	5.93
BLOWEYC	2002	4002	-1.5246E-02	1.5728E-12	2003	2000	0	2	6.09
BQPGABIM	0	50	-3.7903E-05	0.0000E+00	41	36	0	1	0.00
BQPGASIM	0	50	-5.5198E-05	0.0000E+00	45	40	0	1	0.00
BQPGAUSS	0	2003	-3.6258E-01	8.4349E-17	2236	1909	0	1	6.33
CHENHARK	0	5000	-2.0000E+00	0.0000E+00	6003	3000	1	2	7.30
CVXQP1	5000	10000	1.0870E+08	6.9982E-11	11491	1261	0	121	16.88
CVXQP2	2500	10000	8.1842E+07	2.5263E-13	8298	2210	1	25	26.65
DIXON3DQ	0	10000	-4.4409E-16	0.0000E+00	2003	10000	1	2	8.01
DQDRTIC	0	5000	0.0000E+00	0.0000E+00	2003	5000	1	2	5.59
DTOC3	9998	14999	2.3526E+02	1.7762E-15	2004	4999	1	3	13.02
DUAL1	1	85	3.5013E-02	1.3715E-16	77	62	0	2	0.01
DUAL2	1	96	3.3734E-02	1.5379E-16	94	91	0	2	0.01
DUAL3	1	111	1.3576E-01	2.8691E-16	111	96	0	2	0.01
DUAL4	1	75	7.4609E-01	2.2421E-16	64	61	0	2	0.00
GOULDQP3	9999	19999	2.3796E-05	1.2512E-15	5899	4988	3	19	83.16
GRIDNETA	6724	13284	3.0498E+02	4.6555E-14	2083	2218	1	4	8.36
GRIDNETB	6724	13284	1.4332E+02	6.0692E-14	2003	6561	1	3	8.71
GRIDNETC	6724	13284	1.4832E+02	5.4771E-15	3930	4533	2	4	18.38
HILBERTA	0	10	9.6007E-09	0.0000E+00	8	7	0	1	0.00
HILBERTB	0	50	5.2590E-28	9.5890E-16	51	50	0	1	0.00
HUES-MOD	2	10000	3.4824E+07	1.3105E-12	3828	9444	1	3	8.40
HUESTIS	2	10000	3.4824E+11	1.4321E-11	3827	9444	1	3	8.40
JNLBRNG1	0	15625	-1.8058E-01	0.0000E+00	4020	10247	2	2	18.32
JNLBRNG2	0	15625	-4.1496E+00	0.0000E+00	3639	9139	2	2	15.65
JNLBRNGA	0	15625	-2.6851E-01	0.0000E+00	9971	9968	8	2	41.11
JNLBRNGB	0	15625	-6.2807E+00	0.0000E+00	8480	8477	7	2	34.24
MOSARQP1	700	2500	-3.8214E+03	6.6779E-11	3254	1021	0	23	3.68
MOSARQP2	700	2500	-5.0526E+03	4.4410E-14	2553	1640	0	6	4.06
NOBNDTOR	0	14884	-4.4054E-01	0.0000E+00	5899	12078	4	2	28.68
OBSTCLAE	0	15625	1.9010E+00	0.0000E+00	9182	7950	7	2	65.88
OBSTCLAL	0	15625	1.9010E+00	0.0000E+00	7952	7949	6	2	31.86
OBSTCLBL	0	15625	7.2958E+00	0.0000E+00	17171	11317	12	2	72.04
OBSTCLBM	0	15625	7.2958E+00	0.0000E+00	6977	11317	3	2	34.11
OBSTCLBU	0	15625	7.2958E+00	0.0000E+00	13094	11317	10	2	51.01
ODNAMUR	0	11130	9.2366E+03	5.2750E-16	4686	5512	2	2	642.78
OSLBQP	0	8	6.2500E+00	0.0000E+00	7	6	0	1	0.00
PALMER1C	0	8	9.7605E-02	0.0000E+00	9	8	0	2	0.00
PALMER1D	0	7	6.5267E-01	2.0708E-16	8	7	0	1	0.00
PALMER2C	0	8	1.4369E-02	4.1276E-17	9	8	0	2	0.00
PALMER3C	0	8	1.9538E-02	8.0822E-17	9	8	0	2	0.00
PALMER4C	0	8	5.0311E-02	2.4247E-16	9	8	0	2	0.00
PRIMAL3	111	745	-1.3576E-01	6.5629E-13	711	571	0	3	0.30
PRIMAL4	75	1489	-7.4609E-01	5.2297E-16	1223	1140	0	2	1.18
RDW2D51U	65025	132098	8.3606E-04	8.3450E-09	2514	65025	1	5	403.09
RDW2D52U^m	65025	132098	1.2268E-02	3.1503E-13	2241	65420	1	32	507.61
SOSQP2	10001	20000	-4.9987E+03	1.6293E-07	15238	4982	2	48	75.51
STCQP1	4095	8193	3.6710E+05	2.4805E-16	3570	5717	1	19	9.12
STCQP2	4095	8193	3.7189E+04	3.8666E-16	5799	3970	1	67	9.68
STNQP1	4095	8193	-3.1170E+05	2.6521E-16	4008	5277	1	19	9.43
STNQP2	4095	8193	-5.7497E+05	3.1721E-16	6612	2640	1	67	10.15
TESTQUAD	0	5000	0.0000E+00	0.0000E+00	2003	5000	1	2	5.66
TOINTQOR	0	50	1.1755E+03	2.8758E-16	51	50	0	1	0.00
TORSION1	0	14884	-4.2570E-01	0.0000E+00	9987	9984	8	2	46.28
TORSION2	0	14884	-4.2570E-01	0.0000E+00	6460	9984	5	2	45.71
TORSION3	0	14884	-1.2122E+00	0.0000E+00	5211	5208	4	2	21.36

Table 6: Results for SQIC with UMFPACK on ‘‘large’’CUTEr QPs (continued)

Name	m	n	Objective	$\|Ax-b\|$	Itn	nS	bkFac	nFac	Time
TORSION4	0	14884	-1.2122E+00	0.0000E+00	11399	5208	9	2	76.46
TORSION5	0	14884	-2.8588E+00	0.0000E+00	2571	2568	1	2	9.05
TORSION6	0	14884	-2.8588E+00	0.0000E+00	15017	2568	12	3	90.14
TORSIONA	0	14884	-4.1842E-01	0.0000E+00	10115	10112	8	2	47.00
TORSIONB	0	14884	-4.1842E-01	0.0000E+00	6292	10112	5	2	44.63
TORSIONC	0	14884	-1.2045E+00	0.0000E+00	5275	5272	4	2	21.41
TORSIOND	0	14884	-1.2045E+00	0.0000E+00	11298	5272	9	2	75.99
TORSIONE	0	14884	-2.8508E+00	0.0000E+00	2603	2600	1	2	9.13
TORSIONF	0	14884	-2.8508E+00	0.0000E+00	14909	2600	12	3	89.20
TRIDIA	0	10000	-8.8818E-16	5.5728E-17	2003	10000	1	2	12.06
UBH1	12000	18009	1.1160E+00	4.0247E-13	5880	5997	1	90	28.06
WALL10	0	1461	-4.5595E+05	8.6179E-16	1434	1101	0	16	2.57
WALL100	0	149624	-8.9544E+03	3.1570E-17	6113	110712	4	2	1210.48
WALL20	0	5924	-5.2210E+06	0.0000E+00	3034	4277	4	2	13.96
WALL50	0	37311	-9.5450E+06	2.0090E-18	3585	26961	2	2	74.83
ZANGWIL2	0	2	-1.8200E+01	9.7205E-16	3	2	0	1	0.00

Table 7: Results for SQIC in block-LU mode with LUSOL on ‘‘small’’ CUTEr QPs

Name	m	n	Objective	$\|Ax-b\|$	Itn	nS	bkFac	nFac	Time
A0ENDNDL	15002	45006	0.0000E+00	1.3983E-11	7276	0	1	140	30.54
A0ENINDL	15002	45006	0.0000E+00	1.8889E-13	7226	0	3	140	38.57
A0ENSNDL	15002	45006	-2.2339E-09	3.1829E-09	5781	0	7	105	50.42
A0ESDNDL	15002	45006	0.0000E+00	2.2987E-13	7192	0	4	128	42.52
A0ESINDL	15002	45006	0.0000E+00	5.1829E-12	7165	0	1	139	29.87
A0ESSNDL	15002	45006	-2.3387E-09	5.4459E-09	5865	0	5	98	43.58
A0NNDNDL	20004	60012	0.0000E+00	8.4704E-13	61929	0	45	961	727.56
A0NNDNIL	20004	60012	4.8858E+01	4.3955E-12	12155	61	5	238	91.24
A0NNDNSL	20004	60012	-6.6901E-09	5.3091E-12	39329	0	160	647	1449.15
A0NNSNSL	20004	60012	-2.0139E-09	5.9322E-12	21120	0	72	366	654.03
A0NSDSDL	20004	60012	0.0000E+00	4.3836E-11	30797	0	4	618	181.17
A0NSDSDS	2004	6012	-3.8599E-09	2.0915E-07	1615	0	64	27	5.61
A0NSDSIL	20004	60012	0.0000E+00	4.5080E-13	13753	0	26	273	267.87
A0NSDSSL	20004	60012	-9.0444E-11	1.7863E-12	23318	0	17	469	233.61
A0NSSSSL	20004	60012	-1.5190E-08	6.3321E-12	17173	0	40	338	380.30
A2ENDNDL	15002	45006	0.0000E+00	3.4421E-14	7117	0	39	132	191.41
A2ENINDL	15002	45006	1.2643E-29	5.3049E-14	7114	2	16	129	92.89
A2ENSNDL	15002	45006	-2.9789E-09	6.1110E-12	5568	1	21	96	108.36
A2ESDNDL	15002	45006	2.4947E-26	6.0648E-14	6975	8	60	119	282.50
A2ESINDL	15002	45006	7.3514E-27	9.0896E-14	7009	4	126	130	564.74
A2ESSNDL	15002	45006	-2.9789E-09	6.1356E-12	5422	2	11	93	66.96
A2NNDNDL	20004	60012	-7.0324E-15	7.8094E-08	71943	5	142	1042	1586.34
A2NNDNIL^i	20004	60012	0.0000E+00	6.4845E-14	11717	0	0	241	47.11
A2NNDNSL	20004	60012	-5.1569E-09	7.2703E-12	44025	0	136	645	1315.82
A2NNSNSL	20004	60012	-2.5421E-09	7.4080E-08	22765	0	3	451	108.66
A2NSDSDL	20004	60012	-2.7288E-12	6.3848E-08	50809	8	322	806	2941.85
A2NSDSIL	20004	60012	5.3679E+01	7.4629E-08	15050	253	18	247	210.86
A2NSDSSL	20004	60012	-3.5094E-12	1.7296E-11	29609	0	346	541	2882.75
A2NSSSSL	20004	60012	-8.7014E-05	2.1586E-07	20498	0	19	406	227.55
A5ENDNDL	15002	45006	9.9036E-27	3.8496E-09	6606	18	4	115	40.94
A5ENINDL	15002	45006	2.9018E-31	7.7338E-09	6613	3	10	116	66.73
A5ENSNDL	15002	45006	-1.1557E-09	6.1105E-12	5111	0	18	89	93.97
A5ESDNDL	15002	45006	6.2723E-27	3.2334E-08	6441	7	47	109	224.00
A5ESINDL	15002	45006	1.0319E-27	3.3676E-14	6430	3	49	113	236.94
A5ESSNDL	15002	45006	-1.1311E-09	1.4404E-11	4785	0	32	86	146.21
A5NNDNDL	20004	60012	3.8603E+01	2.7429E-13	64922	183	426	1000	3777.15
A5NNDNIL^i	20004	60012	0.0000E+00	9.3408E-14	10259	0	0	211	42.23
A5NNDNSL^t	20004	60012	1.6745E+00	1.8650E-13	41860	27	609	601	5025.54

Table 7: Results for SQIC in block-LU mode with LUSOL on ‘‘small’’ CUTEr QPs (continued)

Name	m	n	Objective	$\|Ax-b\|$	Itn	nS	bkFac	nFac	Time
A5NNSNSL	20004	60012	-2.2027E-09	2.6288E-07	25815	0	58	406	576.26
A5NSDSDL^t	20004	60012	2.6375E+04	1.7192E-11	41967	0	587	756	5058.75
A5NSDSDM	2004	6012	-3.8599E-09	2.0915E-07	1615	0	64	27	5.58
A5NSDSIL	20004	60012	1.1413E+01	2.5155E-07	11945	614	6	192	108.28
A5NSDSSL^r	20004	60012	4.4189E+04	1.6185E-13	29762	1	463	516	3825.71
A5NSSNSM	2004	6012	-3.8599E-09	2.0915E-07	1615	0	64	27	5.62
A5NSSSSL	20004	60012	-7.8284E-09	3.2229E-07	22517	0	169	365	1436.19
AVGASA	10	8	-4.6319E+00	8.4581E-12	11	3	1	1	0.00
AVGASB	10	8	-4.4832E+00	1.3672E-16	11	3	1	1	0.00
BIGGSC4	7	4	-2.4375E+01	9.8000E-12	12	1	1	1	0.00
BLOCKQP1	5001	10010	-4.9940E+03	1.7816E-16	5014	9	1	2	4.27
BLOCKQP2	5001	10010	-4.9928E+03	3.0600E-08	7516	9	11	2	38.81
BLOCKQP3	5001	10010	-2.4950E+03	1.7785E-16	5014	9	1	2	4.04
BLOCKQP4	5001	10010	-2.4933E+03	3.0600E-08	8493	9	11	2	48.54
BLOCKQP5	5001	10010	-2.4950E+03	1.7785E-16	5020	9	1	2	4.05
BQP1VAR	0	1	0.0000E+00	0.0000E+00	2	0	1	1	0.00
CVXBQP1	0	10000	2.2502E+06	1.8646E-17	10001	0	1	1	6.71
CVXQP3	7500	10000	1.1571E+08	2.4506E-12	12397	411	128	150	1021.38
DEGENQP	8010	20	0.0000E+00	0.0000E+00	12	0	1	2	0.03
DUALC1	215	9	6.1553E+03	2.9271E-15	5	2	1	2	0.00
DUALC2	229	7	3.5513E+03	2.6820E-15	4	2	1	2	0.00
DUALC5	278	8	4.2723E+02	9.8919E-16	5	4	1	2	0.00
DUALC8	503	8	1.8309E+04	1.8361E-15	7	2	1	2	0.00
FERRISDC	210	2200	0.0000E+00	0.0000E+00	1	0	1	2	0.27
GENHS28	8	10	9.2717E-01	8.0709E-15	3	2	1	2	0.00
GMNCASE1	300	175	2.6697E-01	3.0353E-16	102	53	1	1	0.04
GMNCASE2	1050	175	-9.9444E-01	3.6771E-15	104	46	1	1	0.05
GMNCASE3	1050	175	1.5251E+00	1.2018E-14	107	48	1	1	0.05
GMNCASE4	350	175	5.9469E+03	7.7433E-12	141	0	1	1	0.12
GOULDQP1	17	32	-3.4853E+03	1.2249E-11	23	0	1	2	0.00
GOULDQP2	9999	19999	1.8512E-12	1.4803E-16	1	0	1	2	1.77
HARKERP2	0	1000	-5.0000E-01	0.0000E+00	1000	0	1	1	3.24
HATFLDH	7	4	-2.4500E+01	0.0000E+00	4	0	1	1	0.00
HS118	17	15	6.6482E+02	1.2250E-11	23	0	1	1	0.00
HS21	1	2	-9.9960E+01	1.7764E-15	2	1	1	1	0.00
HS268	5	5	-5.4570E-12	1.2897E-15	11	5	1	1	0.00
HS3	0	2	0.0000E+00	0.0000E+00	3	1	1	1	0.00
HS35	1	3	1.1111E-01	3.4648E-11	6	2	1	1	0.00
HS35I	1	3	1.1111E-01	3.4648E-11	6	2	1	1	0.00
HS35MOD	1	3	2.5000E-01	0.0000E+00	2	1	1	1	0.00
HS3MOD	0	2	3.9443E-31	0.0000E+00	3	1	1	1	0.00
HS44	6	4	-1.3000E+01	2.6585E-16	3	0	1	1	0.00
HS44NEW	6	4	-1.3000E+01	3.9878E-16	6	0	1	1	0.00
HS51	3	5	-8.8818E-16	1.5632E-16	3	2	1	2	0.00
HS52	3	5	5.3266E+00	1.1236E-16	3	2	1	2	0.00
HS53	3	5	4.0930E+00	1.5632E-16	3	2	1	2	0.00
HS76	3	4	-4.6818E+00	3.0304E-16	5	2	1	1	0.00
HS76I	3	4	-4.6818E+00	3.0304E-16	5	2	1	1	0.00
KSIP	1001	20	5.7580E-01	5.5634E-16	1544	18	1212	19	9.17
LEUVEN1	2220	1530	-1.5243E+07	7.2855E-09	1516	12	23	18	0.96
LEUVEN2	2329	1530	-1.4147E+07	1.6953E-08	610	2	3	9	0.20
LEUVEN3^r	2973	1200	1.4515E+06	8.0334E-11	324	20	63	22	7.37
LEUVEN4	2973	1200	-1.4083E+09	2.7000E-11	1563	50	57	5	10.56
LEUVEN5^r	2973	1200	1.4515E+06	8.0334E-11	324	20	63	22	7.37
LEUVEN6	3091	1200	-1.4533E+08	1.1978E-09	580	30	31	6	5.96
LEUVEN7	946	360	6.9455E+02	2.4889E-12	203	19	7	1	0.21
LINCONT^i	419	1257	0.0000E+00	4.6663E-14	126	0	0	4	0.03
LISWET1	10000	10002	3.6121E+01	4.0546E-12	4	2	2	1	0.98
LISWET10	10000	10002	4.9483E+01	2.8787E-12	46	16	10	1	4.43
LISWET11	10000	10002	4.9524E+01	1.9980E-12	41	30	11	1	4.85

Table 7: Results for SQIC in block-LU mode with LUSOL on ‘‘small’’ CUTEr QPs (continued)

Name	m	n	Objective	$\|Ax-b\|$	Itn	nS	bkFac	nFac	Time
LISWET12	10000	10002	1.7369E+03	1.1531E-12	23	5	12	1	5.28
LISWET2	10000	10002	2.5000E+01	9.8184E-13	20	4	6	1	2.71
LISWET3	10000	10002	2.5000E+01	2.8630E-08	440	261	12	1	5.92
LISWET4	10000	10002	2.5000E+01	2.0362E-07	429	270	12	1	5.88
LISWET5	10000	10002	2.5000E+01	4.9305E-08	411	254	7	1	3.76
LISWET6	10000	10002	2.5000E+01	1.2886E-12	340	222	7	1	3.63
LISWET7	10000	10002	4.9884E+02	5.1492E-13	4	2	3	1	1.41
LISWET8	10000	10002	7.1447E+02	4.4402E-13	21	13	6	1	2.72
LISWET9	10000	10002	1.9632E+03	1.1087E-12	16	4	10	1	4.37
LOTSCHD	7	12	2.3984E+03	3.4710E-15	8	0	1	2	0.00
MARATOSB^{u}	0	2	-1.4400E+06	2.0016E-17	2	0	1	1	0.00
MPC1	3833	2550	-2.3262E+07	2.0091E-08	1361	0	10	17	1.10
MPC10	2351	1530	-1.5034E+07	4.8004E-08	1122	11	2	14	0.83
MPC11	2351	1530	-1.5030E+07	9.1769E-09	930	34	1	11	0.71
MPC12	2351	1530	-1.5033E+07	1.9357E-07	1152	19	5	13	0.66
MPC13	2351	1530	-1.5034E+07	4.9519E-09	1075	13	5	13	0.65
MPC14	2351	1530	-1.5034E+07	3.1434E-07	1189	16	1	14	0.95
MPC15	2351	1530	-1.5034E+07	2.1451E-06	1000	15	4	12	0.56
MPC16	2351	1530	-1.5034E+07	8.5864E-07	1054	16	5	12	0.48
MPC2	2351	1530	-1.5033E+07	9.2528E-08	1201	27	32	13	0.75
MPC3	2351	1530	-1.5030E+07	2.3069E-08	1220	32	64	15	0.99
MPC4	2351	1530	-1.5033E+07	4.7028E-09	1246	21	2	15	0.59
MPC5	2351	1530	-1.5033E+07	9.2840E-08	1233	25	3	16	0.58
MPC6	2351	1530	-1.5034E+07	4.7094E-09	1201	18	11	13	0.64
MPC7	2351	1530	-1.5034E+07	1.6580E-09	1079	15	5	13	0.72
MPC8	2351	1530	-1.5034E+07	5.1513E-09	1070	13	3	12	0.76
MPC9	2351	1530	-1.5034E+07	1.9323E-07	1150	11	3	13	0.72
NASH^{i}	24	72	0.0000E+00	9.4369E-16	2	0	0	2	0.00
NCVXBQP1	0	10000	-1.9855E+10	0.0000E+00	10015	0	1	1	7.00
NCVXBQP2	0	10000	-1.3245E+10	1.8604E-17	11223	51	2	1	7.86
NCVXBQP3	0	10000	-6.4122E+09	2.3813E-15	10837	126	3	1	7.35
NCVXQP1	500	1000	-7.1562E+07	6.8959E-11	749	0	5	4	0.16
NCVXQP2	500	1000	-5.7759E+07	4.0210E-11	1064	0	8	4	0.35
NCVXQP3	500	1000	-2.9253E+07	3.7099E-11	1221	19	17	3	0.41
NCVXQP4	250	1000	-9.3995E+07	1.0040E-11	788	0	1	2	0.14
NCVXQP5	250	1000	-6.6257E+07	1.0750E-11	824	0	1	2	0.17
NCVXQP6	250	1000	-3.4172E+07	6.9414E-12	926	49	2	2	0.33
NCVXQP7	750	1000	-4.3521E+07	7.1410E-11	674	0	2	8	0.21
NCVXQP8	750	1000	-3.0103E+07	9.6844E-11	919	0	9	9	0.43
NCVXQP9	750	1000	-2.1230E+07	2.0935E-09	947	11	9	8	0.47
PENTDI	0	5000	-7.5000E-01	0.0000E+00	3	2	1	1	0.07
PORTSNQP	2	100000	3.3332E+04	3.5313E-14	108265	257	4	882	42.73
PORTSQP	1	100000	3.3331E+04	1.8823E-11	100317	315	4	2	23.48
POWELL20	10000	10000	5.2090E+10	2.6350E-08	5003	1	11	1	26.62
PRIMAL1	85	325	-3.5013E-02	1.5669E-12	218	133	3	1	0.05
PRIMAL2	96	649	-3.3734E-02	1.5035E-14	408	302	4	1	0.12
PRIMALC1	9	230	-6.1553E+03	2.9296E-12	19	14	1	1	0.00
PRIMALC2	7	231	-3.5513E+03	3.4245E-13	4	1	1	1	0.00
PRIMALC5	8	287	-4.2723E+02	1.8948E-14	10	5	1	1	0.00
PRIMALC8	8	520	-1.8309E+04	1.6012E-10	22	17	1	1	0.00
QPBAND	5000	10000	-9.9992E+03	1.8821E-11	29960	39	19	1	118.57
QPCBLEND	74	83	-7.8425E-03	1.5447E-09	76	2	7	2	0.01
QPCBOEI1	351	384	1.1504E+07	1.4995E-10	702	113	9	6	0.10
QPCBOEI2	166	143	8.1720E+06	5.0484E-11	210	32	5	3	0.01
QPCSTAIR	356	467	6.2044E+06	2.6688E-12	303	21	27	5	0.10
QPNBAND	5000	10000	-4.9997E+04	9.6494E-16	15000	1	10	1	41.14
QPNBLEND	74	83	-8.7056E-03	1.8434E-11	72	3	4	2	0.00
QPNBOEI1	351	384	6.7367E+06	8.0460E-10	684	92	10	7	0.08
QPNBOEI2	166	143	1.3683E+06	5.7738E-11	229	27	7	4	0.01
QPNSTAIR	356	467	5.1460E+06	1.1944E-11	349	20	21	5	0.08

Table 7: Results for SQIC in block-LU mode with LUSOL on ‘‘small’’ CUTEr QPs (continued)

Name	m	n	Objective	$\|Ax-b\|$	Itn	nS	bkFac	nFac	Time
QUDLIN	0	5000	-1.2500E+09	0.0000E+00	5000	0	1	1	0.72
RDW2D51F	65025	132098	1.1353E-03	2.9003E-07	2258	0	1	33	1678.92
RDW2D52F	49	162	8.6159E-03	1.3838E-15	71	37	1	3	0.01
S268	5	5	-5.4570E-12	1.2897E-15	11	5	1	1	0.00
SIM2BQP	0	2	0.0000E+00	0.0000E+00	2	0	1	1	0.00
SIMBQP	0	2	3.4667E-31	0.0000E+00	3	1	1	1	0.00
SOSQP1	10001	20000	-2.4500E-11	1.5765E-11	3	0	1	2	1.36
STATIC3^u	96	434	-3.0892E+02	0.0000E+00	3	1	1	2	0.00
STEENBRA	108	432	1.6958E+04	8.1667E-11	87	11	1	2	0.01
TAME	1	2	0.0000E+00	6.0309E-17	2	1	1	2	0.00
YAO	2000	2002	1.9770E+02	1.9799E-15	3	1	2	1	0.05
ZECEVIC2	2	2	-4.1250E+00	0.0000E+00	3	1	1	1	0.00

Table 8: Results for SQIC in block-LU mode with LUSOL on ‘‘large’’ CUTEr QPs

Name	m	n	Objective	$\|Ax-b\|$	Itn	nS	bkFac	nFac	Time
ALLINQP	25000	50000	-5.4813E+03	1.4836E-14	24100	9820	17	2	325.24
AUG2D	10000	20200	1.6874E+06	2.2694E-11	10193	10192	10	2	82.91
AUG2DC	10000	20200	1.8184E+06	3.4459E-10	10201	10200	13	2	73.51
AUG2DCQP	10000	20200	6.4981E+06	1.5187E-11	14516	9994	19	6	104.36
AUG2DQP	10000	20200	6.2370E+06	1.6333E-11	14314	9801	17	6	102.91
AUG3D	8000	27543	2.4561E+04	2.1023E-10	16910	16909	17	2	159.71
AUG3DC	8000	27543	2.7654E+04	1.9784E-10	19544	19543	23	2	235.51
AUG3DCQP	8000	27543	6.1560E+04	7.3420E-08	22185	17665	24	17	206.51
AUG3DQP	8000	27543	5.4229E+04	1.6890E-11	18457	13712	20	17	149.14
BIGGSB1	0	5000	1.5000E-02	6.5035E-17	5004	4998	11	1	8.06
BLOWEYA	2002	4002	-2.2781E-02	4.8741E-14	2002	2000	2	2	6.07
BLOWEYB	2002	4002	-1.5226E-02	1.0456E-13	2002	2000	2	2	6.07
BLOWEYC	2002	4002	-1.5246E-02	1.5033E-12	2003	2000	2	2	6.40
BQPGABIM	0	50	-3.7903E-05	0.0000E+00	41	36	1	1	0.00
BQPGASIM	0	50	-5.5198E-05	0.0000E+00	45	40	1	1	0.00
BQPGAUSS	0	2003	-3.6258E-01	2.0033E-16	2236	1909	8	1	1.79
CHENHARK	0	5000	-2.0000E+00	2.6282E-17	6998	2997	9	1	3.73
CVXQP1	5000	10000	1.0870E+08	9.1305E-11	11347	1261	100	71	176.04
CVXQP2	2500	10000	8.1842E+07	7.2588E-12	8444	2209	7	6	29.14
DIXON3DQ	0	10000	0.0000E+00	1.1706E-15	10001	10000	16	1	26.24
DQDRTIC	0	5000	0.0000E+00	0.0000E+00	5001	5000	11	1	7.03
DTOC3	9998	14999	2.3526E+02	3.1364E-10	5001	4999	21	2	153.40
DUAL1	1	85	3.5013E-02	1.5200E-16	77	62	2	2	0.01
DUAL2	1	96	3.3734E-02	1.9737E-16	94	91	2	2	0.01
DUAL3	1	111	1.3576E-01	2.9847E-16	111	96	2	2	0.01
DUAL4	1	75	7.4609E-01	2.4499E-16	64	61	2	2	0.01
GOULDQP3	9999	19999	2.3796E-05	1.9257E-15	5897	4988	7	2	31.40
GRIDNETA	6724	13284	3.0498E+02	5.5533E-10	2250	2183	3	2	9.25
GRIDNETB	6724	13284	1.4332E+02	2.9715E-12	6562	6561	7	2	52.90
GRIDNETC	6724	13284	1.4832E+02	6.3700E-10	5219	4533	5	3	33.67
HILBERTA	0	10	9.6007E-09	0.0000E+00	8	7	1	1	0.00
HILBERTB	0	50	5.2957E-28	9.5890E-16	51	50	2	1	0.00
HUES-MOD	2	10000	3.4824E+07	5.8496E-13	10632	9441	15	2	29.39
HUESTIS	2	10000	3.4824E+11	4.1544E-12	10630	9440	15	2	29.38
JNLBRNG1	0	15625	-1.8058E-01	0.0000E+00	10248	10247	16	1	37.68
JNLBRNG2	0	15625	-4.1496E+00	0.0000E+00	9140	9139	15	1	30.50
JNLBRNGA	0	15625	-2.6851E-01	0.0000E+00	9969	9968	16	1	36.37
JNLBRNGB	0	15625	-6.2807E+00	0.0000E+00	8478	8477	14	1	27.68
MOSARQP1	700	2500	-3.8214E+03	6.4742E-11	3253	1021	66	1	3.84
MOSARQP2	700	2500	-5.0526E+03	6.0674E-13	2553	1640	3	1	3.21
NOBNDTOR	0	14884	-4.4054E-01	0.0000E+00	12083	12078	18	1	50.39
OBSTCLAE	0	15625	1.9010E+00	0.0000E+00	22296	7950	27	1	106.30
OBSTCLAL	0	15625	1.9010E+00	0.0000E+00	7950	7949	14	1	25.71

Table 8: Results for SQIC in block-LU mode with LUSOL on ‘‘large’’ CUTEr QPs (continued)

Name	m	n	Objective	$\|Ax-b\|$	Itn	nS	bkFac	nFac	Time
OBSTCLBL	0	15625	7.2958E+00	0.0000E+00	17169	11317	20	1	59.82
OBSTCLBM	0	15625	7.2958E+00	0.0000E+00	19846	11317	22	1	58.22
OBSTCLBU	0	15625	7.2958E+00	0.0000E+00	13092	11317	19	1	41.91
ODNAMUR	0	11130	9.2366E+03	2.1100E-15	6273	4548	11	1	649.48
OSLBQP	0	8	6.2500E+00	0.0000E+00	7	6	1	1	0.00
PALMER1C	0	8	9.7605E-02	9.0073E-17	9	8	1	1	0.00
PALMER1D	0	7	6.5267E-01	8.2833E-17	8	7	1	1	0.00
PALMER2C	0	8	1.4369E-02	1.2383E-16	9	8	1	1	0.00
PALMER3C	0	8	1.9538E-02	8.0822E-17	9	8	1	1	0.00
PALMER4C	0	8	5.0311E-02	0.0000E+00	9	8	1	1	0.00
PRIMAL3	111	745	-1.3576E-01	6.5625E-13	712	572	4	1	0.32
PRIMAL4	75	1489	-7.4609E-01	7.1580E-17	1223	1140	6	1	0.69
RDW2D51U^t	65025	132098	1.3373E+03	1.4738E-13	489	359	76	4	5396.37
RDW2D52U^t	65025	132098	1.2197E-02	1.3247E-12	24440	24211	36	2	5001.43
SOSQP2	10001	20000	-4.9987E+03	9.6974E-12	15876	4976	10	2	55.05
STCQP1	4095	8193	3.6710E+05	1.4555E-12	7277	5717	6	18	18.83
STCQP2	4095	8193	3.7189E+04	3.1792E-16	7694	3970	4	66	15.46
STNQP1	4095	8193	-3.1170E+05	3.3473E-12	7284	5277	6	18	17.59
STNQP2	4095	8193	-5.7497E+05	2.1518E-12	7251	2640	3	66	9.40
TESTQUAD	0	5000	0.0000E+00	0.0000E+00	5001	5000	11	1	7.01
TOINTQOR	0	50	1.1755E+03	1.4379E-16	51	50	2	1	0.00
TORSION1	0	14884	-4.2570E-01	0.0000E+00	9985	9984	16	1	40.27
TORSION2	0	14884	-4.2570E-01	0.0000E+00	18818	9984	24	1	91.63
TORSION3	0	14884	-1.2122E+00	0.0000E+00	5209	5208	11	1	14.84
TORSION4	0	14884	-1.2122E+00	0.0000E+00	23594	5208	29	1	102.30
TORSION5	0	14884	-2.8588E+00	0.0000E+00	2569	2568	9	1	4.97
TORSION6	0	14884	-2.8588E+00	0.0000E+00	26235	2568	33	2	97.69
TORSIONA	0	14884	-4.1842E-01	0.0000E+00	10113	10112	16	1	40.55
TORSIONB	0	14884	-4.1842E-01	0.0000E+00	18690	10112	24	1	90.57
TORSIONC	0	14884	-1.2045E+00	0.0000E+00	5273	5272	11	1	15.14
TORSIOND	0	14884	-1.2045E+00	0.0000E+00	23530	5272	29	1	102.24
TORSIONE	0	14884	-2.8508E+00	0.0000E+00	2601	2600	9	1	4.94
TORSIONF	0	14884	-2.8508E+00	0.0000E+00	26203	2600	33	2	95.16
TRIDIA	0	10000	-8.8818E-16	3.0447E-13	10000	9999	16	1	25.84
UBH1	12000	18009	1.1160E+00	1.4812E-07	9901	5997	1865	14	79.07
WALL10	0	1461	-4.5595E+05	3.5875E-19	1434	1101	13	1	0.67
WALL100^t	0	149624	-2.6946E+03	1.4350E-19	58540	58289	63	1	5083.28
WALL20	0	5924	-5.2210E+06	1.4350E-19	5389	4276	26	6	15.92
WALL50	0	37311	-9.5450E+06	6.4862E-17	39999	26958	58	1	734.66
ZANGWIL2	0	2	-1.8200E+01	9.7205E-16	3	2	1	1	0.00

Table 9: Results for SQIC in block-LU mode with HSL_MA57 on ‘‘small’’ CUTEr QPs

Name	m	n	Objective	$\|Ax-b\|$	Itn	nS	bkFac	nFac	Time
A0ENDNDL	15002	45006	0.0000E+00	1.2326E-13	7266	0	1	140	28.17
A0ENINDL	15002	45006	0.0000E+00	1.8700E-13	7220	0	1	140	28.13
A0ENSNDL	15002	45006	-2.1776E-09	3.1783E-09	5781	0	96	105	617.87
A0ESDNDL	15002	45006	0.0000E+00	7.5620E-15	6824	0	1	128	27.58
A0ESINDL	15002	45006	0.0000E+00	1.5804E-13	7162	0	1	139	27.86
A0ESSNDL	15002	45006	-2.2589E-09	5.4328E-09	5865	0	145	98	807.75
A0NNDNDL	20004	60012	0.0000E+00	2.1841E-12	49589	0	10	961	328.69
A0NNDNIL	20004	60012	7.8883E+01	2.4738E-13	12107	76	26	238	443.20
A0NNDNSL	20004	60012	-6.6963E-09	5.3036E-12	39329	0	207	647	1848.40
A0NNSNSL	20004	60012	-2.0284E-09	5.9326E-12	21120	0	107	366	1075.64
A0NSDSDL	20004	60012	0.0000E+00	3.5429E-14	30910	0	44	620	460.66
A0NSDSDS	2004	6012	-3.8600E-09	2.0915E-07	1615	0	129	27	11.08
A0NSDSIL	20004	60012	4.7445E+00	1.5503E-13	13512	38	22	274	415.24
A0NSDSSL	20004	60012	-9.6676E-11	4.0045E-13	23318	0	14	469	380.33

Table 9: Results for SQIC in block-LU mode with HSL_MA57 on ''small'' CUTEr QPs (continued)

Name	m	n	Objective	$\|Ax-b\|$	Itn	nS	bkFac	nFac	Time
A0NSSSSL	20004	60012	-1.5069E-08	6.3224E-12	17173	0	36	338	544.63
A2ENDNDL	15002	45006	1.4488E-31	2.7992E-14	6804	48	1	132	26.99
A2ENINDL	15002	45006	1.3495E-31	1.0767E-13	6704	56	1	129	26.51
A2ENSNDL	15002	45006	-2.9982E-09	9.3846E-12	5568	1	137	96	777.15
A2ESDNDL	15002	45006	2.4667E-30	1.1928E-13	6328	75	1	119	26.68
A2ESINDL	15002	45006	1.6337E-30	5.6514E-14	6690	44	1	130	26.48
A2ESSNDL	15002	45006	-2.9646E-09	6.0830E-12	5422	2	236	93	1139.45
A2NNDNDL	20004	60012	4.7384E-10	7.8724E-08	54681	53	15	1039	392.70
A2NNDNIL^{i}	20004	60012	0.0000E+00	6.4845E-14	11717	0	0	241	46.87
A2NNDNSL	20004	60012	-5.1896E-09	4.5866E-12	44025	0	189	645	1834.02
A2NNSNSL	20004	60012	-2.5389E-09	7.4080E-08	22765	0	7	451	364.92
A2NSDSDL	20004	60012	-1.1892E-11	1.0949E-07	40439	17	100	803	907.75
A2NSDSIL	20004	60012	6.0896E+01	5.5122E-08	13575	254	24	247	440.84
A2NSDSSL	20004	60012	-5.1042E-12	1.8121E-12	29609	0	352	541	2806.10
A2NSSSSL	20004	60012	-8.7014E-05	2.1586E-07	20498	0	27	406	498.56
A5ENDNDL	15002	45006	2.3796E-31	1.4387E-13	5904	229	1	115	24.43
A5ENINDL	15002	45006	5.2628E-31	1.3525E-13	5914	222	1	115	24.05
A5ENSNDL	15002	45006	-1.1462E-09	3.8909E-11	5111	0	83	89	559.37
A5ESDNDL	15002	45006	8.1125E-31	2.0240E-13	5673	239	1	109	24.02
A5ESINDL	15002	45006	3.7442E-29	1.8582E-13	5755	197	1	112	24.19
A5ESSNDL	15002	45006	-1.1655E-09	6.1750E-12	4785	0	110	86	654.96
A5NNDNDL	20004	60012	5.8115E-09	1.1283E-07	51582	166	20	999	418.27
A5NNDNIL^{i}	20004	60012	0.0000E+00	9.3408E-14	10259	0	0	211	41.08
A5NNDNSL	20004	60012	9.1500E-14	1.9457E-13	41976	0	595	601	4634.92
A5NNSNSL	20004	60012	-2.2883E-09	2.6288E-07	25628	0	102	406	1097.72
A5NSDSDL	20004	60012	-7.3361E-12	1.9192E-13	38232	31	94	742	871.09
A5NSDSDM	2004	6012	-3.8600E-09	2.0915E-07	1615	0	129	27	11.11
A5NSDSIL	20004	60012	1.0008E+01	2.5155E-07	11697	602	18	192	405.40
A5NSDSSL^{r}	20004	60012	4.4189E+04	1.5155E-13	29762	1	426	516	3349.46
A5NSSNSM	2004	6012	-3.8600E-09	2.0915E-07	1615	0	129	27	11.02
A5NSSSSL	20004	60012	-7.8836E-09	3.2229E-07	22517	0	189	365	1684.04
AVGASA	10	8	-4.6319E+00	8.4581E-12	11	3	1	1	0.00
AVGASB	10	8	-4.4832E+00	1.5039E-16	11	3	1	1	0.00
BIGGSC4	7	4	-2.4375E+01	9.8000E-12	12	1	1	1	0.00
BLOCKQP1	5001	10010	-4.9940E+03	1.7788E-16	5014	9	2	2	6.07
BLOCKQP2	5001	10010	-4.9938E+03	2.0015E-07	5007	9	7	2	126.85
BLOCKQP3	5001	10010	-2.4950E+03	1.7759E-16	5014	9	2	2	6.29
BLOCKQP4	5001	10010	-2.4958E+03	6.1284E-11	7402	9	11	2	152.00
BLOCKQP5	5001	10010	-2.4950E+03	1.7768E-16	5020	9	2	2	5.96
BQP1VAR	0	1	0.0000E+00	1.0503E-17	2	0	1	1	0.00
CVXBQP1	0	10000	2.2502E+06	0.0000E+00	10008	0	18	1	52.05
CVXQP3^{r}	7500	10000	1.8605E+08	3.7145E-10	8985	0	24	164	459.64
DEGENQP	8010	20	-1.4884E-13	1.4251E-16	12	0	1	2	0.07
DUALC1	215	9	6.1553E+03	2.0490E-15	5	2	1	2	0.00
DUALC2	229	7	3.5513E+03	2.6820E-15	4	2	1	2	0.00
DUALC5	278	8	4.2723E+02	7.6091E-16	5	4	1	2	0.00
DUALC8	503	8	1.8309E+04	2.1421E-15	7	2	1	2	0.00
FERRISDC	210	2200	-1.4835E-05	3.3607E-15	188	96	1	2	0.68
GENHS28	8	10	9.2717E-01	2.5610E-14	1	2	1	2	0.00
GMNCASE1	300	175	2.6697E-01	6.0117E-16	54	95	3	1	0.14
GMNCASE2	1050	175	-9.9444E-01	9.0061E-16	56	94	1	1	0.08
GMNCASE3	1050	175	1.5251E+00	2.2186E-16	60	93	2	1	0.10
GMNCASE4	350	175	5.9469E+03	7.7433E-12	141	0	1	1	0.29
GOULDQP1	17	32	-3.4853E+03	1.2249E-11	24	0	4	2	0.00
GOULDQP2	9999	19999	1.8512E-12	0.0000E+00	1	0	1	2	1.17
HARKERP2	0	1000	-5.0000E-01	7.1418E-16	1000	1	4	1	6.47
HATFLDH	7	4	-2.4500E+01	9.8000E-12	7	0	3	1	0.00
HS118	17	15	6.6482E+02	1.4797E-12	16	0	1	1	0.00
HS21	1	2	-9.9960E+01	1.7764E-15	1	1	1	1	0.00
HS268	5	5	-1.8190E-11	3.2243E-15	3	4	1	1	0.00

Table 9: Results for SQIC in block-LU mode with HSL_MA57 on ''small'' CUTEr QPs (continued)

Name	m	n	Objective	$\|Ax-b\|$	Itn	nS	bkFac	nFac	Time
HS3	0	2	0.0000E+00	0.0000E+00	1	2	1	1	0.00
HS35	1	3	1.1111E-01	1.1102E-16	2	2	1	1	0.00
HS35I	1	3	1.1111E-01	1.1102E-16	2	2	1	1	0.00
HS35MOD	1	3	2.5000E-01	0.0000E+00	1	2	1	1	0.00
HS3MOD	0	2	0.0000E+00	0.0000E+00	2	1	3	1	0.00
HS44	6	4	-1.3000E+01	2.6585E-16	3	0	1	1	0.00
HS44NEW	6	4	-1.3000E+01	3.9878E-16	6	0	3	1	0.00
HS51	3	5	2.6645E-15	1.5632E-16	1	2	1	2	0.00
HS52	3	5	5.3266E+00	1.1382E-15	1	2	1	2	0.00
HS53	3	5	4.0930E+00	5.0804E-16	1	2	1	2	0.00
HS76	3	4	-4.6818E+00	1.6288E-15	5	2	1	1	0.00
HS76I	3	4	-4.6818E+00	1.6288E-15	5	2	1	1	0.00
KSIP	1001	20	5.7580E-01	3.1639E-19	157	18	124	1	1.43
LEUVEN1	2220	1530	-1.5243E+07	7.2945E-09	1516	14	24	18	2.65
LEUVEN2	2329	1530	-1.4147E+07	1.6934E-08	617	4	9	10	0.78
LEUVEN3^r	2973	1200	1.5516E+06	3.9315E-09	22	0	22	22	6.75
LEUVEN4	2973	1200	-1.4083E+09	1.6890E-11	1560	50	95	12	55.26
LEUVEN5^r	2973	1200	1.5516E+06	3.9315E-09	22	0	22	22	6.75
LEUVEN6	3091	1200	-1.4515E+08	3.9054E-10	617	29	71	7	33.26
LEUVEN7	946	360	6.9455E+02	2.4889E-12	137	21	1	1	0.60
LINCONT^i	419	1257	0.0000E+00	4.6663E-14	126	0	0	4	0.03
LISWET1	10000	10002	3.6121E+01	1.8540E-11	1	2	1	1	0.48
LISWET10	10000	10002	4.9483E+01	2.9606E-16	21	17	9	1	3.35
LISWET11	10000	10002	4.9524E+01	3.5897E-15	43	36	11	1	4.17
LISWET12	10000	10002	1.7369E+03	1.4803E-16	24	6	16	1	5.76
LISWET2	10000	10002	2.5000E+01	3.3677E-15	17	4	9	1	3.35
LISWET3	10000	10002	2.5000E+01	2.8630E-08	437	261	17	1	12.02
LISWET4	10000	10002	2.5000E+01	2.0362E-07	426	270	18	1	11.21
LISWET5	10000	10002	2.5000E+01	4.9305E-08	408	254	18	1	10.96
LISWET6	10000	10002	2.5000E+01	1.7393E-15	337	222	17	1	10.06
LISWET7	10000	10002	4.9884E+02	1.3334E-10	1	2	1	1	0.49
LISWET8	10000	10002	7.1447E+02	5.5511E-16	18	14	9	1	3.33
LISWET9	10000	10002	1.9632E+03	7.4015E-17	14	5	11	1	4.00
LOTSCHD	7	12	2.3984E+03	5.5536E-15	8	0	1	2	0.00
MARATOSB^u	0	2	-1.4400E+06	2.0016E-17	2	2	2	1	0.00
MPC1	3833	2550	-2.3262E+07	1.6583E-08	1378	0	7	17	2.61
MPC10	2351	1530	-1.5034E+07	4.8005E-08	1177	11	12	14	1.88
MPC11	2351	1530	-1.5030E+07	6.8429E-10	977	34	24	12	1.77
MPC12	2351	1530	-1.5033E+07	1.9330E-07	1148	19	3	13	1.74
MPC13	2351	1530	-1.5034E+07	4.9523E-09	1106	13	22	13	2.34
MPC14	2351	1530	-1.5034E+07	3.1434E-07	1204	16	8	14	1.71
MPC15	2351	1530	-1.5034E+07	2.1455E-06	1024	15	8	12	1.58
MPC16	2351	1530	-1.5034E+07	8.5848E-07	1061	16	2	12	1.34
MPC2	2351	1530	-1.5033E+07	9.2533E-08	1236	27	23	13	2.23
MPC3	2351	1530	-1.5030E+07	2.3069E-08	1259	32	39	15	2.39
MPC4	2351	1530	-1.5033E+07	1.0657E-07	1302	21	16	16	2.01
MPC5	2351	1530	-1.5033E+07	9.2772E-08	1285	25	6	16	1.29
MPC6	2351	1530	-1.5034E+07	4.6960E-09	1227	18	35	13	2.57
MPC7	2351	1530	-1.5034E+07	2.9264E-09	1134	15	7	13	1.56
MPC8	2351	1530	-1.5034E+07	1.0453E-09	1165	13	8	13	2.02
MPC9	2351	1530	-1.5034E+07	1.9312E-07	1201	11	3	13	2.17
NASH^i	24	72	0.0000E+00	9.4369E-16	2	0	0	2	0.00
NCVXBQP1	0	10000	-1.9855E+10	0.0000E+00	10015	0	2	1	7.96
NCVXBQP2	0	10000	-1.3245E+10	1.8604E-17	11223	51	3	1	8.82
NCVXBQP3	0	10000	-6.4122E+09	2.3813E-15	10837	126	5	1	11.24
NCVXQP1	500	1000	-7.1562E+07	6.7744E-11	749	0	15	4	0.59
NCVXQP2	500	1000	-5.7752E+07	5.7163E-10	940	0	6	4	1.09
NCVXQP3	500	1000	-3.0766E+07	4.0933E-10	1158	0	122	5	8.16
NCVXQP4	250	1000	-9.3995E+07	1.0313E-11	788	0	3	2	0.18
NCVXQP5	250	1000	-6.6257E+07	1.0799E-11	824	0	3	2	0.24

Table 9: Results for SQIC in block-LU mode with HSL_MA57 on ''small'' CUTEr QPs (continued)

Name	m	n	Objective	$\|Ax-b\|$	Itn	nS	bkFac	nFac	Time
NCVXQP6	250	1000	-3.5087E+07	3.0351E-12	1173	49	127	2	3.88
NCVXQP7	750	1000	-4.3521E+07	8.2087E-11	653	0	29	8	1.70
NCVXQP8	750	1000	-3.0103E+07	6.8226E-09	960	0	25	9	2.12
NCVXQP9	750	1000	-2.1230E+07	2.6650E-10	948	11	48	9	2.99
PENTDI	0	5000	-7.5000E-01	0.0000E+00	3	2	1	1	0.07
PORTSNQP	2	100000	3.3332E+04	2.8979E-16	108265	257	4	882	42.74
PORTSQP	1	100000	3.3331E+04	5.7105E-16	100315	315	4	2	23.48
POWELL20	10000	10000	5.2090E+10	1.2968E-10	5001	1	5	1	33.01
PRIMAL1	85	325	-3.5013E-02	1.3743E-09	71	262	1	1	0.03
PRIMAL2	96	649	-3.3734E-02	1.8528E-17	97	557	1	1	0.05
PRIMALC1	9	230	-6.1553E+03	5.1268E-12	5	14	1	1	0.00
PRIMALC2	7	231	-3.5513E+03	2.2830E-13	4	1	1	1	0.00
PRIMALC5	8	287	-4.2723E+02	8.8423E-14	6	5	1	1	0.00
PRIMALC8	8	520	-1.8309E+04	1.5284E-10	8	17	1	1	0.00
QPBAND	5000	10000	-9.9992E+03	1.8821E-11	29960	39	19	1	605.98
QPCBLEND	74	83	-7.8425E-03	1.5447E-09	76	2	12	2	0.01
QPCBOEI1	351	384	1.1504E+07	1.0624E-09	702	113	18	6	0.22
QPCBOEI2	166	143	8.1720E+06	2.0576E-10	214	32	14	3	0.03
QPCSTAIR	356	467	6.2044E+06	2.6575E-12	303	21	40	5	0.18
QPNBAND	5000	10000	-4.9997E+04	9.6494E-16	15000	1	10	1	56.33
QPNBLEND	74	83	-8.7056E-03	1.8434E-11	72	3	6	2	0.01
QPNBOEI1	351	384	6.7367E+06	7.8558E-10	684	92	18	7	0.17
QPNBOEI2	166	143	1.3683E+06	5.7827E-11	229	27	37	4	0.05
QPNSTAIR	356	467	5.1460E+06	1.1943E-11	349	20	23	5	0.13
QUDLIN	0	5000	-1.2500E+09	0.0000E+00	5000	0	1	1	0.72
RDW2D51F	65025	132098	1.1353E-03	2.9003E-07	2258	0	1	33	1595.18
RDW2D52F	49	162	8.6159E-03	1.9115E-15	71	37	1	3	0.01
S268	5	5	-1.8190E-11	3.2243E-15	3	4	1	1	0.00
SIM2BQP	0	2	0.0000E+00	0.0000E+00	2	0	1	1	0.00
SIMBQP	0	2	3.4667E-31	0.0000E+00	2	1	1	1	0.00
SOSQP1	10001	20000	-6.9026E-11	3.7303E-14	3	9999	1	2	2.03
STATIC3^u	96	434	-2.5298E+03	9.5504E-11	14	256	2	2	0.01
STEENBRA	108	432	1.6958E+04	8.1667E-11	85	11	1	2	0.02
TAME	1	2	0.0000E+00	6.0309E-17	2	1	1	2	0.00
YAO	2000	2002	1.9770E+02	5.5511E-17	3	1	3	1	0.07
ZECEVIC2	2	2	-4.1250E+00	0.0000E+00	3	1	1	1	0.00

Table 10: Results for SQIC in block-LU mode with HSL_MA57 on ''large'' CUTEr QPs

Name	m	n	Objective	$\|Ax-b\|$	Itn	nS	bkFac	nFac	Time
ALLINQP	25000	50000	-5.4813E+03	3.9804E-15	16957	9820	10	2	312.62
AUG2D	10000	20200	1.6874E+06	9.0949E-13	1	10200	1	2	24.85
AUG2DC	10000	20200	1.8184E+06	1.8415E-12	1	10200	1	2	1.82
AUG2DCQP	10000	20200	6.4981E+06	4.6268E-12	14422	9994	18	6	146.27
AUG2DQP	10000	20200	6.2370E+06	1.6333E-11	14411	9801	18	6	143.84
AUG3D	8000	27543	2.4561E+04	2.0507E-11	16910	16909	18	2	295.12
AUG3DC	8000	27543	2.7654E+04	6.1995E-13	1	19543	1	2	3.55
AUG3DCQP	8000	27543	6.1560E+04	7.3399E-08	22095	17665	23	17	250.18
AUG3DQP	8000	27543	5.4229E+04	1.6334E-11	18403	13712	20	17	190.74
BIGGSB1	0	5000	1.5000E-02	0.0000E+00	5003	4998	11	1	9.36
BLOWEYA	2002	4002	-2.2781E-02	2.4997E-14	1205	2000	2	2	4.66
BLOWEYB	2002	4002	-1.5226E-02	4.2000E-13	805	2000	1	2	2.52
BLOWEYC	2002	4002	-1.5246E-02	8.8006E-13	805	2000	1	2	2.50
BQPGABIM	0	50	-3.7903E-05	0.0000E+00	11	36	1	1	0.00
BQPGASIM	0	50	-5.5198E-05	5.1482E-21	11	40	1	1	0.00
BQPGAUSS	0	2003	-3.6258E-01	1.6870E-16	265	1909	4	1	0.32
CHENHARK	0	5000	-2.0000E+00	7.1890E-12	2010	2992	2	1	28.65
CVXQP1^r	5000	10000	1.0870E+08	3.2506E-12	12913	1244	376	85	1935.46

Table 10: Results for SQIC in block-LU mode with HSL_MA57 on ''large'' CUTEr QPs (continued)

Name	m	n	Objective	$\|Ax-b\|$	Itn	nS	bkFac	nFac	Time
CVXQP2	2500	10000	8.1842E+07	6.8493E-12	6346	2210	6	6	164.39
DIXON3DQ	0	10000	0.0000E+00	0.0000E+00	1	10000	1	1	0.89
DQDRTIC	0	5000	0.0000E+00	0.0000E+00	1	5000	1	1	0.12
DTOC3	9998	14999	2.3526E+02	2.3683E-15	2	4999	1	2	1.02
DUAL1	1	85	3.5013E-02	1.3335E-16	77	62	2	2	0.01
DUAL2	1	96	3.3734E-02	1.2688E-16	94	91	2	2	0.01
DUAL3	1	111	1.3576E-01	1.5413E-16	111	96	2	2	0.01
DUAL4	1	75	7.4609E-01	1.9161E-16	64	61	2	2	0.01
GOULDQP3	9999	19999	2.3796E-05	1.9409E-15	5897	4988	7	2	189.73
GRIDNETA	6724	13284	3.0498E+02	1.9551E-10	219	2218	4	2	2.31
GRIDNETB	6724	13284	1.4332E+02	8.6893E-14	1	6561	1	2	0.81
GRIDNETC	6724	13284	1.4832E+02	6.4831E-10	2498	4533	3	3	15.81
HILBERTA^u	0	10	-5.2633E-05	0.0000E+00	2	10	2	1	0.00
HILBERTB	0	50	4.2805E-28	0.0000E+00	1	50	1	1	0.00
HUES-MOD	2	10000	3.4824E+07	6.9025E-12	563	9444	1	2	2.10
HUESTIS	2	10000	3.4824E+11	1.4294E-11	561	9444	1	2	2.10
JNLBRNG1	0	15625	-1.8058E-01	0.0000E+00	2745	10247	3	1	15.66
JNLBRNG2	0	15625	-4.1496E+00	0.0000E+00	1637	9139	2	1	9.00
JNLBRNGA	0	15625	-2.6851E-01	0.0000E+00	9969	9968	16	1	41.07
JNLBRNGB	0	15625	-6.2807E+00	0.0000E+00	8478	8477	14	1	31.07
MOSARQP1	700	2500	-3.8214E+03	3.4055E-07	1498	1021	2	1	3.16
MOSARQP2	700	2500	-5.0526E+03	9.1905E-08	851	1640	1	1	1.61
NOBNDTOR	0	14884	-4.4054E-01	0.0000E+00	4879	12078	5	1	28.94
OBSTCLAE	0	15625	1.9010E+00	0.0000E+00	7180	7950	7	1	61.60
OBSTCLAL	0	15625	1.9010E+00	0.0000E+00	7950	7949	14	1	29.68
OBSTCLBL	0	15625	7.2958E+00	0.0000E+00	17169	11317	20	1	74.05
OBSTCLBM	0	15625	7.2958E+00	0.0000E+00	3814	11317	4	1	31.02
OBSTCLBU	0	15625	7.2958E+00	0.0000E+00	13092	11317	19	1	50.39
ODNAMUR	0	11130	9.2366E+03	2.6375E-16	3600	5512	3	1	436.01
OSLBQP	0	8	6.2500E+00	0.0000E+00	1	6	1	1	0.00
PALMER1C	0	8	9.7605E-02	1.8015E-17	1	8	1	1	0.00
PALMER1D	0	7	6.5267E-01	4.1417E-17	1	7	1	1	0.00
PALMER2C	0	8	1.4369E-02	1.2383E-16	1	8	1	1	0.00
PALMER3C	0	8	1.9538E-02	8.0822E-17	1	8	1	1	0.00
PALMER4C	0	8	5.0311E-02	1.6164E-16	1	8	1	1	0.00
PRIMAL3	111	745	-1.3576E-01	4.2872E-17	102	648	1	1	0.10
PRIMAL4	75	1489	-7.4609E-01	8.6188E-17	63	1427	1	1	0.07
RDW2D51U	65025	132098	8.3606E-04	8.3437E-09	512	65025	1	4	281.58
RDW2D52U	65025	132098	1.1373E-02	2.6378E-08	512	65025	1	2	315.75
SOSQP2	10001	20000	-4.9987E+03	7.5992E-07	19409	4985	15	2	794.36
STCQP1	4095	8193	3.6710E+05	1.4554E-12	1551	5717	1	18	10.60
STCQP2	4095	8193	3.7189E+04	3.5229E-16	3279	3970	1	66	2.93
STNQP1	4095	8193	-3.1170E+05	3.3471E-12	2134	5277	32	18	51.89
STNQP2	4095	8193	-5.7497E+05	1.8398E-14	4875	2640	523	67	628.95
TESTQUAD	0	5000	0.0000E+00	9.4814E-17	1	5000	1	1	0.11
TOINTQOR	0	50	1.1755E+03	0.0000E+00	1	50	1	1	0.00
TORSION1	0	14884	-4.2570E-01	0.0000E+00	9985	9984	16	1	45.70
TORSION2	0	14884	-4.2570E-01	0.0000E+00	4418	9984	5	1	39.83
TORSION3	0	14884	-1.2122E+00	0.0000E+00	5209	5208	11	1	16.61
TORSION4	0	14884	-1.2122E+00	0.0000E+00	9194	5208	9	1	72.52
TORSION5	0	14884	-2.8588E+00	0.0000E+00	2569	2568	9	1	5.37
TORSION6	0	14884	-2.8588E+00	0.0000E+00	11834	2568	12	1	87.17
TORSIONA	0	14884	-4.1842E-01	0.0000E+00	10113	10112	16	1	46.29
TORSIONB	0	14884	-4.1842E-01	0.0000E+00	4290	10112	5	1	39.44
TORSIONC	0	14884	-1.2045E+00	0.0000E+00	5273	5272	11	1	17.32
TORSIOND	0	14884	-1.2045E+00	0.0000E+00	9130	5272	9	1	71.78
TORSIONE	0	14884	-2.8508E+00	0.0000E+00	2601	2600	9	1	5.39
TORSIONF	0	14884	-2.8508E+00	0.0000E+00	11802	2600	12	1	86.91
TRIDIA	0	10000	-1.1102E-15	0.0000E+00	1	10000	1	1	0.99
UBH1	12000	18009	4.8981E+01	1.2312E-07	1091	290	33	15	117.62

Table 10: Results for SQIC in block-LU mode with HSL_MA57 on ''large'' CUTEr QPs (continued)

Name	m	n	Objective	$\|Ax-b\|$	Itn	nS	bkFac	nFac	Time
WALL10	0	1461	-4.5595E+05	1.4350E-19	70	1101	2	1	0.10
WALL100	0	149624	-8.9544E+03	2.5256E-17	5889	110712	6	1	1636.22
WALL20	0	5924	-5.2210E+06	0.0000E+00	118	4277	2	1	1.11
WALL50	0	37311	-9.5450E+06	4.0180E-18	858	26961	6	1	85.71
ZANGWIL2	0	2	-1.8200E+01	9.7205E-16	1	2	1	1	0.00

Table 11: Results for SQIC in block-LU mode with UMFPACK on ''small'' CUTEr QPs

Name	m	n	Objective	$\|Ax-b\|$	Itn	nS	bkFac	nFac	Time
A0ENDNDL	15002	45006	0.0000E+00	3.2578E-13	7276	0	1	140	31.17
A0ENINDL	15002	45006	0.0000E+00	3.4807E-13	7226	0	3	140	39.24
A0ENSNDL	15002	45006	-2.2296E-09	3.1834E-09	5781	0	7	105	50.86
A0ESDNDL	15002	45006	0.0000E+00	1.5132E-13	7192	0	4	128	43.36
A0ESINDL	15002	45006	0.0000E+00	5.3479E-13	7165	0	1	139	29.86
A0ESSNDL	15002	45006	-2.2516E-09	5.4169E-09	5865	0	4	98	40.86
A0NNDNDL	20004	60012	0.0000E+00	1.9500E-12	61933	0	41	961	745.41
A0NNDNIL	20004	60012	4.8858E+01	1.1409E-13	12155	61	7	238	109.97
A0NNDNSL	20004	60012	-6.6953E-09	5.3029E-12	39329	0	171	647	1561.80
A0NNSNSL	20004	60012	-2.0108E-09	5.9323E-12	21120	0	82	366	731.36
A0NSDSDL	20004	60012	0.0000E+00	7.0008E-13	30797	0	8	618	216.00
A0NSDSDS	2004	6012	-3.8599E-09	2.0915E-07	1615	0	64	27	5.86
A0NSDSIL	20004	60012	0.0000E+00	1.0632E-13	13753	0	29	273	298.50
A0NSDSSL	20004	60012	-9.4087E-11	4.0037E-13	23318	0	16	469	221.16
A0NSSSSL	20004	60012	-1.5083E-08	6.3287E-12	17173	0	33	338	324.99
A2ENDNDL	15002	45006	0.0000E+00	7.5928E-15	7117	0	39	132	194.67
A2ENINDL	15002	45006	1.9866E-32	3.7407E-14	7114	2	17	129	99.13
A2ENSNDL	15002	45006	-2.9766E-09	6.1113E-12	5568	1	17	96	92.42
A2ESDNDL	15002	45006	1.3371E-36	3.2265E-15	6975	8	60	119	286.50
A2ESINDL	15002	45006	4.5918E-41	5.7976E-15	7009	4	127	130	577.06
A2ESSNDL	15002	45006	-2.9773E-09	6.1114E-12	5422	2	11	93	65.23
A2NNDNDL	20004	60012	5.8227E-11	5.9117E-08	71836	19	344	1041	3297.71
A2NNDNIL^i	20004	60012	0.0000E+00	6.4845E-14	11717	0	0	241	46.88
A2NNDNSL	20004	60012	-3.8611E-09	3.4160E-12	44024	0	152	645	1467.34
A2NNSNSL	20004	60012	-2.5432E-09	7.4080E-08	22765	0	3	451	109.19
A2NSDSDL	20004	60012	-2.7288E-12	6.3848E-08	50809	8	332	806	3085.36
A2NSDSIL	20004	60012	5.3679E+01	7.4629E-08	15050	253	4	247	110.39
A2NSDSSL	20004	60012	1.7602E-11	1.6463E-12	29609	0	359	541	3010.87
A2NSSSSL	20004	60012	-8.7014E-05	2.1586E-07	20498	0	20	406	236.05
A5ENDNDL	15002	45006	1.7993E-27	3.8496E-09	6606	18	4	115	41.69
A5ENINDL	15002	45006	4.1424E-31	7.7338E-09	6613	3	10	116	67.48
A5ENSNDL	15002	45006	-1.1172E-09	6.1114E-12	5111	0	28	89	132.19
A5ESDNDL	15002	45006	6.8652E-32	3.2334E-08	6441	7	47	109	228.29
A5ESINDL	15002	45006	8.8346E-36	1.2402E-14	6430	3	49	113	236.39
A5ESSNDL	15002	45006	-1.1307E-09	6.1750E-12	4785	0	48	86	206.01
A5NNDNDL^t	20004	60012	5.5772E+02	7.1479E-13	63924	1	565	1000	5039.97
A5NNDNIL^i	20004	60012	0.0000E+00	9.3408E-14	10259	0	0	211	41.38
A5NNDNSL^t	20004	60012	1.5486E+02	1.7961E-13	41240	3	600	601	5015.55
A5NNSNSL	20004	60012	-3.4835E-09	2.6288E-07	25628	0	28	406	353.86
A5NSDSDL^t	20004	60012	2.6720E+04	6.4192E-14	41947	0	573	756	5010.14
A5NSDSDM	2004	6012	-3.8599E-09	2.0915E-07	1615	0	64	27	5.86
A5NSDSIL	20004	60012	1.1413E+01	2.5155E-07	11945	614	7	192	123.58
A5NSDSSL^r	20004	60012	4.4189E+04	7.4266E-14	29762	1	467	516	3888.93
A5NSSNSM	2004	6012	-3.8599E-09	2.0915E-07	1615	0	64	27	5.83
A5NSSSSL	20004	60012	-7.6536E-09	3.2229E-07	22516	0	110	365	973.34
AVGASA	10	8	-4.6319E+00	8.4581E-12	11	3	1	1	0.00
AVGASB	10	8	-4.4832E+00	8.2033E-17	11	3	1	1	0.00
BIGGSC4	7	4	-2.4375E+01	9.8000E-12	12	1	1	1	0.00
BLOCKQP1	5001	10010	-4.9940E+03	1.7805E-16	5014	9	1	2	4.29

Table 11: Results for SQIC in block-LU mode with UMFPACK on ''small'' CUTEr QPs (continued)

Name	m	n	Objective	$\|Ax-b\|$	Itn	nS	bkFac	nFac	Time
BLOCKQP2	5001	10010	-4.9928E+03	3.0600E-08	7516	9	11	2	42.07
BLOCKQP3	5001	10010	-2.4950E+03	1.7785E-16	5014	9	1	2	4.32
BLOCKQP4	5001	10010	-2.4933E+03	3.0600E-08	8493	9	11	2	50.12
BLOCKQP5	5001	10010	-2.4950E+03	1.7785E-16	5020	9	1	2	4.31
BQP1VAR	0	1	0.0000E+00	0.0000E+00	2	0	1	1	0.00
CVXBQP1	0	10000	2.2502E+06	1.8646E-17	10001	0	1	1	6.77
CVXQP3^{r**}	7500	10000	2.5916E+08	8.2832E-09	7085	0	48	164	1199.69
DEGENQP	8010	20	-1.5701E-15	5.5933E-19	12	0	1	2	0.03
DUALC1	215	9	6.1553E+03	4.0980E-15	5	2	1	2	0.00
DUALC2	229	7	3.5513E+03	2.4138E-15	4	2	1	2	0.00
DUALC5	278	8	4.2723E+02	1.1414E-15	5	4	1	2	0.00
DUALC8	503	8	1.8309E+04	1.5301E-15	7	2	1	2	0.00
FERRISDC	210	2200	1.8323E-27	1.8444E-21	1	0	1	2	0.27
GENHS28	8	10	9.2717E-01	1.0089E-14	3	2	1	2	0.00
GMNCASE1	300	175	2.6697E-01	1.5980E-16	102	53	2	1	0.04
GMNCASE2	1050	175	-9.9444E-01	8.9043E-16	104	46	1	1	0.05
GMNCASE3	1050	175	1.5251E+00	1.3604E-16	107	48	1	1	0.05
GMNCASE4	350	175	5.9469E+03	7.7433E-12	141	0	1	1	0.12
GOULDQP1	17	32	-3.4853E+03	1.2249E-11	23	0	1	2	0.00
GOULDQP2	9999	19999	1.8512E-12	0.0000E+00	1	0	1	2	1.81
HARKERP2	0	1000	-5.0000E-01	0.0000E+00	1000	0	1	1	3.25
HATFLDH	7	4	-2.4500E+01	0.0000E+00	4	0	1	1	0.00
HS118	17	15	6.6482E+02	1.2250E-11	23	0	1	1	0.00
HS21	1	2	-9.9960E+01	1.7764E-15	2	1	1	1	0.00
HS268	5	5	-3.6380E-12	1.9346E-15	11	5	1	1	0.00
HS3	0	2	0.0000E+00	0.0000E+00	3	1	1	1	0.00
HS35	1	3	1.1111E-01	3.4648E-11	6	2	1	1	0.00
HS35I	1	3	1.1111E-01	3.4648E-11	6	2	1	1	0.00
HS35MOD	1	3	2.5000E-01	2.2204E-16	2	1	1	1	0.00
HS3MOD	0	2	3.9443E-31	0.0000E+00	3	1	1	1	0.00
HS44	6	4	-1.3000E+01	2.6585E-16	3	0	1	1	0.00
HS44NEW	6	4	-1.3000E+01	3.9878E-16	6	0	1	1	0.00
HS51	3	5	8.8818E-16	3.9080E-17	3	2	1	2	0.00
HS52	3	5	5.3266E+00	1.1236E-16	3	2	1	2	0.00
HS53	3	5	4.0930E+00	7.8160E-17	3	2	1	2	0.00
HS76	3	4	-4.6818E+00	3.0304E-16	5	2	1	1	0.00
HS76I	3	4	-4.6818E+00	3.0304E-16	5	2	1	1	0.00
KSIP	1001	20	5.7580E-01	2.0732E-19	1544	18	1213	19	3.08
LEUVEN1	2220	1530	-1.5243E+07	7.2851E-09	1516	12	10	18	1.43
LEUVEN2	2329	1530	-1.4147E+07	1.7186E-08	610	2	1	9	0.33
LEUVEN3^{r}	2973	1200	1.4515E+06	6.2258E-15	324	20	63	22	3.68
LEUVEN4	2973	1200	-1.4083E+09	2.7000E-11	1571	50	76	5	7.29
LEUVEN5^{r}	2973	1200	1.4515E+06	6.2258E-15	324	20	63	22	3.67
LEUVEN6	3091	1200	-1.4533E+08	1.1978E-09	579	30	30	5	3.17
LEUVEN7	946	360	6.9455E+02	2.4889E-12	203	19	7	1	0.19
LINCONT^{i}	419	1257	0.0000E+00	4.6663E-14	126	0	0	4	0.03
LISWET1	10000	10002	3.6121E+01	2.2204E-16	4	2	3	1	1.42
LISWET10	10000	10002	4.9483E+01	4.6629E-15	46	16	11	1	4.97
LISWET11	10000	10002	4.9524E+01	1.4248E-14	41	30	13	1	5.91
LISWET12	10000	10002	1.7369E+03	2.5905E-16	23	5	15	1	6.70
LISWET2	10000	10002	2.5000E+01	4.8110E-16	20	4	6	1	2.76
LISWET3	10000	10002	2.5000E+01	2.8630E-08	440	261	11	1	8.94
LISWET4	10000	10002	2.5000E+01	2.0362E-07	429	270	13	1	9.36
LISWET5	10000	10002	2.5000E+01	4.9305E-08	411	254	7	1	6.65
LISWET6	10000	10002	2.5000E+01	1.6653E-15	340	222	8	1	6.26
LISWET7	10000	10002	4.9884E+02	7.4015E-17	4	2	3	1	1.42
LISWET8	10000	10002	7.1447E+02	4.4409E-16	21	13	8	1	3.63
LISWET9	10000	10002	1.9632E+03	7.4015E-17	16	4	11	1	4.90
LOTSCHD	7	12	2.3984E+03	5.5536E-15	8	0	1	2	0.00
MARATOSB^{u}	0	2	-1.4400E+06	2.0016E-17	2	0	1	1	0.00

Table 11: Results for SQIC in block-LU mode with UMFPACK on ''small'' CUTEr QPs (continued)

Name	m	n	Objective	$\|Ax - b\|$	Itn	nS	bkFac	nFac	Time
MPC1	3833	2550	-2.3262E+07	1.5626E-08	1364	0	3	17	1.59
MPC10	2351	1530	-1.5034E+07	4.8004E-08	1123	11	4	14	0.88
MPC11	2351	1530	-1.5030E+07	9.1770E-09	924	34	6	11	0.86
MPC12	2351	1530	-1.5033E+07	1.9342E-07	1154	19	4	13	1.48
MPC13	2351	1530	-1.5034E+07	4.9521E-09	1073	13	3	13	0.97
MPC14	2351	1530	-1.5034E+07	3.1434E-07	1187	16	3	14	1.07
MPC15	2351	1530	-1.5034E+07	2.1451E-06	1006	15	3	12	1.00
MPC16	2351	1530	-1.5034E+07	8.5870E-07	1049	16	2	12	0.79
MPC2	2351	1530	-1.5033E+07	9.2528E-08	1201	27	9	13	1.30
MPC3	2351	1530	-1.5030E+07	2.3069E-08	1218	32	42	15	2.03
MPC4	2351	1530	-1.5033E+07	4.7024E-09	1227	21	7	15	1.00
MPC5	2351	1530	-1.5033E+07	9.2840E-08	1266	25	3	16	0.70
MPC6	2351	1530	-1.5034E+07	1.3974E-09	1173	18	13	13	1.38
MPC7	2351	1530	-1.5034E+07	2.9797E-09	1081	15	4	13	0.87
MPC8	2351	1530	-1.5034E+07	5.1577E-09	1074	13	9	12	1.18
MPC9	2351	1530	-1.5034E+07	1.9341E-07	1151	11	6	13	1.06
NASH^{i}	24	72	0.0000E+00	9.4369E-16	2	0	0	2	0.00
NCVXBQP1	0	10000	-1.9855E+10	0.0000E+00	10015	0	1	1	7.05
NCVXBQP2	0	10000	-1.3245E+10	1.8604E-17	11223	51	2	1	7.93
NCVXBQP3	0	10000	-6.4122E+09	2.3813E-15	10837	126	3	1	7.41
NCVXQP1	500	1000	-7.1562E+07	6.8611E-11	749	0	3	4	0.33
NCVXQP2	500	1000	-5.7759E+07	6.9266E-11	1044	0	7	4	1.22
NCVXQP3	500	1000	-2.8846E+07	4.5112E-11	1187	18	24	3	1.20
NCVXQP4	250	1000	-9.3995E+07	1.0046E-11	788	0	1	2	0.15
NCVXQP5	250	1000	-6.6257E+07	1.0755E-11	824	0	1	2	0.18
NCVXQP6	250	1000	-3.4172E+07	6.9487E-12	926	49	2	2	0.35
NCVXQP7	750	1000	-4.3521E+07	7.0148E-11	653	0	4	8	0.78
NCVXQP8	750	1000	-3.0103E+07	4.7593E-10	953	0	17	9	2.57
NCVXQP9	750	1000	-2.1230E+07	1.8780E-09	943	11	17	8	2.72
PENTDI	0	5000	-7.5000E-01	0.0000E+00	3	2	1	1	0.07
PORTSNQP	2	100000	3.3332E+04	1.6559E-16	108265	257	4	882	42.74
PORTSQP	1	100000	3.3331E+04	1.8823E-11	100317	315	4	2	23.50
POWELL20	10000	10000	5.2090E+10	2.6349E-08	5003	1	11	1	29.43
PRIMAL1	85	325	-3.5013E-02	1.5669E-12	218	133	3	1	0.06
PRIMAL2	96	649	-3.3734E-02	1.6095E-17	408	302	4	1	0.16
PRIMALC1	9	230	-6.1553E+03	2.9296E-12	19	14	1	1	0.00
PRIMALC2	7	231	-3.5513E+03	3.4245E-13	4	1	1	1	0.00
PRIMALC5	8	287	-4.2723E+02	1.8948E-14	10	5	1	1	0.00
PRIMALC8	8	520	-1.8309E+04	1.4637E-10	22	17	1	1	0.00
QPBAND	5000	10000	-9.9992E+03	1.8821E-11	29960	39	19	1	122.48
QPCBLEND	74	83	-7.8425E-03	1.5447E-09	76	2	14	2	0.01
QPCBOEI1	351	384	1.1504E+07	3.9499E-10	701	113	6	6	0.15
QPCBOEI2	166	143	8.1720E+06	5.0188E-11	210	32	13	3	0.02
QPCSTAIR	356	467	6.2044E+06	2.6679E-12	303	21	37	5	0.15
QPNBAND	5000	10000	-4.9997E+04	9.6494E-16	15000	1	10	1	43.98
QPNBLEND	74	83	-8.7056E-03	1.8434E-11	72	3	6	2	0.01
QPNBOEI1	351	384	6.7367E+06	8.0464E-10	675	92	22	7	0.09
QPNBOEI2	166	143	1.3683E+06	5.7715E-11	229	27	12	4	0.02
QPNSTAIR	356	467	5.1460E+06	1.1943E-11	348	21	24	5	0.12
QUDLIN	0	5000	-1.2500E+09	0.0000E+00	5000	0	1	1	0.74
RDW2D51F	65025	132098	1.1353E-03	2.9003E-07	2258	0	1	33	1720.27
RDW2D52F	49	162	8.6159E-03	3.5910E-15	71	37	1	3	0.01
S268	5	5	-3.6380E-12	1.9346E-15	11	5	1	1	0.00
SIM2BQP	0	2	0.0000E+00	0.0000E+00	2	0	1	1	0.00
SIMBQP	0	2	3.4667E-31	0.0000E+00	3	1	1	1	0.00
SOSQP1	10001	20000	-2.4500E-11	1.5765E-11	3	0	1	2	1.34
STATIC3^{u}	96	434	-3.0892E+02	0.0000E+00	3	1	1	2	0.00
STEENBRA	108	432	1.6958E+04	8.1667E-11	87	11	1	2	0.01
TAME	1	2	0.0000E+00	6.0309E-17	2	1	1	2	0.00
YAO	2000	2002	1.9770E+02	5.5511E-17	3	1	3	1	0.07

Table 11: Results for SQIC in block-LU mode with UMFPACK on ''small'' CUTEr QPs (continued)

Name	m	n	Objective	$\|Ax-b\|$	Itn	nS	bkFac	nFac	Time
ZECEVIC2	2	2	-4.1250E+00	0.0000E+00	3	1	1	1	0.00

Table 12: Results for SQIC in block-LU mode with UMFPACK on ''large'' CUTEr QPs

Name	m	n	Objective	$\|Ax-b\|$	Itn	nS	bkFac	nFac	Time
ALLINQP	25000	50000	-5.4813E+03	1.8575E-14	24100	9820	17	2	343.49
AUG2D	10000	20200	1.6874E+06	3.2567E-13	10193	10192	11	2	79.03
AUG2DC	10000	20200	1.8184E+06	3.6987E-11	10201	10200	14	2	81.84
AUG2DCQP	10000	20200	6.4981E+06	5.6654E-12	14315	9994	19	6	110.57
AUG2DQP	10000	20200	6.2370E+06	1.6333E-11	14335	9801	17	6	109.55
AUG3D	8000	27543	2.4561E+04	2.3166E-11	16910	16909	17	2	160.42
AUG3DC	8000	27543	2.7654E+04	3.5081E-11	19544	19543	20	2	220.55
AUG3DCQP	8000	27543	6.1560E+04	7.3399E-08	22078	17665	23	17	200.52
AUG3DQP	8000	27543	5.4229E+04	1.6334E-11	18423	13712	20	17	148.10
BIGGSB1	0	5000	1.5000E-02	0.0000E+00	5004	4998	11	1	8.54
BLOWEYA	2002	4002	-2.2781E-02	6.5365E-14	2002	2000	2	2	15.78
BLOWEYB	2002	4002	-1.5226E-02	1.0795E-13	2002	2000	2	2	15.43
BLOWEYC	2002	4002	-1.5246E-02	1.4235E-12	2003	2000	2	2	16.12
BQPGABIM	0	50	-3.7903E-05	0.0000E+00	41	36	1	1	0.00
BQPGASIM	0	50	-5.5198E-05	0.0000E+00	45	40	1	1	0.00
BQPGAUSS	0	2003	-3.6258E-01	6.3262E-16	2236	1909	8	1	1.93
CHENHARK	0	5000	-2.0000E+00	2.6282E-17	6998	2997	9	1	3.94
CVXQP1**	5000	10000	1.0870E+08	2.4897E-11	18617	1266	151	141	1199.19
CVXQP2	2500	10000	8.1842E+07	7.2585E-12	8444	2209	8	6	43.75
DIXON3DQ	0	10000	4.4409E-16	1.5608E-15	10001	10000	16	1	27.85
DQDRTIC	0	5000	0.0000E+00	0.0000E+00	5001	5000	11	1	7.30
DTOC3	9998	14999	2.3526E+02	1.7762E-15	5001	4999	75	2	95.34
DUAL1	1	85	3.5013E-02	1.3438E-16	77	62	2	2	0.01
DUAL2	1	96	3.3734E-02	1.2688E-16	94	91	2	2	0.01
DUAL3	1	111	1.3576E-01	1.8167E-17	111	96	2	2	0.01
DUAL4	1	75	7.4609E-01	2.3602E-16	64	61	2	2	0.01
GOULDQP3	9999	19999	2.3796E-05	1.9409E-15	5897	4988	7	2	89.23
GRIDNETA	6724	13284	3.0498E+02	5.5533E-10	2250	2183	3	2	10.83
GRIDNETB	6724	13284	1.4332E+02	1.8713E-14	6561	6560	7	2	46.91
GRIDNETC	6724	13284	1.4832E+02	2.0037E-07	5229	4533	5	3	33.57
HILBERTA	0	10	9.6007E-09	0.0000E+00	8	7	1	1	0.00
HILBERTB	0	50	5.2957E-28	9.5890E-16	51	50	2	1	0.00
HUES-MOD	2	10000	3.4824E+07	1.4379E-12	10632	9441	15	2	29.64
HUESTIS	2	10000	3.4824E+11	4.9975E-12	10630	9440	15	2	29.62
JNLBRNG1	0	15625	-1.8058E-01	0.0000E+00	10248	10247	16	1	40.04
JNLBRNG2	0	15625	-4.1496E+00	0.0000E+00	9140	9139	15	1	32.03
JNLBRNGA	0	15625	-2.6851E-01	0.0000E+00	9969	9968	16	1	37.18
JNLBRNGB	0	15625	-6.2807E+00	0.0000E+00	8478	8477	14	1	28.57
MOSARQP1	700	2500	-3.8214E+03	6.4747E-11	3253	1021	71	1	4.14
MOSARQP2	700	2500	-5.0526E+03	2.1591E-13	2553	1640	3	1	3.30
NOBNDTOR	0	14884	-4.4054E-01	0.0000E+00	12083	12078	18	1	53.29
OBSTCLAE	0	15625	1.9010E+00	0.0000E+00	22296	7950	27	1	111.48
OBSTCLAL	0	15625	1.9010E+00	0.0000E+00	7950	7949	14	1	27.77
OBSTCLBL	0	15625	7.2958E+00	0.0000E+00	17169	11317	20	1	66.50
OBSTCLBM	0	15625	7.2958E+00	0.0000E+00	19846	11317	22	1	63.63
OBSTCLBU	0	15625	7.2958E+00	0.0000E+00	13092	11317	19	1	45.76
ODNAMUR	0	11130	9.2366E+03	3.6925E-15	6273	4548	11	1	625.75
OSLBQP	0	8	6.2500E+00	0.0000E+00	7	6	1	1	0.00
PALMER1C	0	8	9.7605E-02	9.0073E-17	9	8	1	1	0.00
PALMER1D	0	7	6.5267E-01	8.2833E-17	8	7	1	1	0.00
PALMER2C	0	8	1.4369E-02	1.2383E-16	9	8	1	1	0.00
PALMER3C	0	8	1.9538E-02	8.0822E-17	9	8	1	1	0.00
PALMER4C	0	8	5.0311E-02	0.0000E+00	9	8	1	1	0.00

Table 12: Results for SQIC in block-LU mode with UMFPACK on ''large'' CUTEr QP (continued)

Name	m	n	Objective	$\|Ax-b\|$	Itn	nS	bkFac	nFac	Time
PRIMAL3	111	745	-1.3576E-01	6.5650E-13	712	572	4	1	0.37
PRIMAL4	75	1489	-7.4609E-01	6.7198E-17	1223	1140	6	1	0.80
RDW2D51U^t	65025	132098	2.6640E+03	1.0057E-14	369	239	3	4	5397.56
RDW2D52U^t	65025	132098	1.2268E-02	2.5456E-13	540	369	2	2	5096.46
SOSQP2	10001	20000	-4.9987E+03	1.2482E-11	15876	4976	10	2	56.32
STCQP1	4095	8193	3.6710E+05	1.4554E-12	7277	5717	6	18	20.45
STCQP2	4095	8193	3.7189E+04	3.8666E-16	7698	3970	4	66	16.26
STNQP1	4095	8193	-3.1170E+05	3.3473E-12	7284	5277	6	18	19.90
STNQP2	4095	8193	-5.7497E+05	2.1518E-12	7251	2640	3	66	10.22
TESTQUAD	0	5000	0.0000E+00	0.0000E+00	5001	5000	11	1	7.18
TOINTQOR	0	50	1.1755E+03	1.4379E-16	51	50	2	1	0.00
TORSION1	0	14884	-4.2570E-01	0.0000E+00	9985	9984	16	1	42.33
TORSION2	0	14884	-4.2570E-01	0.0000E+00	18818	9984	24	1	93.81
TORSION3	0	14884	-1.2122E+00	0.0000E+00	5209	5208	11	1	15.87
TORSION4	0	14884	-1.2122E+00	0.0000E+00	23594	5208	29	1	105.68
TORSION5	0	14884	-2.8588E+00	0.0000E+00	2569	2568	9	1	5.17
TORSION6	0	14884	-2.8588E+00	0.0000E+00	26235	2568	33	2	102.16
TORSIONA	0	14884	-4.1842E-01	0.0000E+00	10113	10112	16	1	42.67
TORSIONB	0	14884	-4.1842E-01	0.0000E+00	18690	10112	24	1	93.91
TORSIONC	0	14884	-1.2045E+00	0.0000E+00	5273	5272	11	1	16.09
TORSIOND	0	14884	-1.2045E+00	0.0000E+00	23530	5272	29	1	106.04
TORSIONE	0	14884	-2.8508E+00	0.0000E+00	2601	2600	9	1	5.19
TORSIONF	0	14884	-2.8508E+00	0.0000E+00	26203	2600	33	2	101.03
TRIDIA	0	10000	-8.8818E-16	1.5166E-13	10000	9999	16	1	27.99
UBH1	12000	18009	1.1160E+00	1.4812E-07	9901	5997	1865	14	169.24
WALL10	0	1461	-4.5595E+05	7.1750E-19	1414	1101	9	1	0.74
WALL100^t	0	149624	-2.6874E+03	3.4440E-18	55400	55149	60	1	5014.98
WALL20	0	5924	-5.2210E+06	2.2960E-18	5384	4276	14	2	16.26
WALL50	0	37311	-9.5450E+06	4.8790E-17	39815	26958	45	1	844.30
ZANGWIL2	0	2	-1.8200E+01	9.7205E-16	3	2	1	1	0.00